



**Catarina Isabel Cartaxo Pinheiro**

Mestrado Integrado em Engenharia Biomédica

## **Algoritmos de Segmentação para Aplicações Biomédicas**

Dissertação para obtenção do Grau de Mestre em  
Engenharia Biomédica

Orientador: José Manuel Fonseca, Professor Auxiliar com Agregação,  
Faculdade de Ciências e Tecnologia da Universidade  
Nova de Lisboa

Co-orientador: Leonardo Martins, Investigador, Faculdade de Ciências e  
Tecnologia da Universidade Nova de Lisboa



FACULDADE DE  
CIÊNCIAS E TECNOLOGIA  
UNIVERSIDADE NOVA DE LISBOA

**Outubro, 2017**





**Catarina Isabel Cartaxo Pinheiro**

Mestrado Integrado em Engenharia Biomédica

## **Algoritmos de Segmentação para Aplicações Biomédicas**

Dissertação para obtenção do Grau de Mestre em  
Engenharia Biomédica

Orientador: José Manuel Fonseca, Professor Auxiliar com Agregação,  
Faculdade de Ciências e Tecnologia da Universidade  
Nova de Lisboa

Co-orientador: Leonardo Martins, Investigador, Faculdade de Ciências e  
Tecnologia da Universidade Nova de Lisboa



FACULDADE DE  
CIÊNCIAS E TECNOLOGIA  
UNIVERSIDADE NOVA DE LISBOA

iii

**Outubro, 2017**



### **Algoritmos de Segmentação para Aplicações Biomédicas**

Copyright © Catarina Isabel Cartaxo Pinheiro, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa.

A Faculdade de Ciências e Tecnologia e a Universidade Nova de Lisboa têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objetivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.



*“Vou ensinar-te e mostrar-te o caminho que deves seguir;  
guiar-te-ei sem te perder de vista.”*

*Salmos 32:8*





# Resumo

Atualmente, um dos maiores desafios associados à área da biologia relaciona-se com a natureza e o comportamento dos mecanismos que regulam o ciclo celular.

O *Laboratory of Biosystem Dynamics*, estabelecido na Finlândia, conta com a colaboração de um grupo de investigação interdisciplinar para realizar múltiplas medições *single cell, in vivo* que servem de suporte a diversos estudos. Um dos principais objetivos destas investigações consiste em testar e propor novas ferramentas computacionais e modelos estocásticos capazes de descrever e possibilitar uma análise mais aprofundada de alguns dos mecanismos que regulam o ciclo celular. Devido à sua simplicidade, fácil proliferação e manipulação, a *Escherichia coli* torna-se o objeto ideal para estudar este tipo de fenómenos. Através da microscopia de contraste de fase, uma técnica amplamente empregue na área da biologia e medicina, é possível observar, com um elevado detalhe, organismos transparentes no seu estado natural.

Em muitos laboratórios de microbiologia, as imagens continuam a ser frequentemente examinadas apenas por inspeção visual, o que pode tornar este processo extremamente subjetivo, demorado e conduzir a conclusões erróneas. Assim sendo, o processamento computacional e a análise de imagens são determinantes para obter informação estatisticamente relevante e fidedigna. Em muitas aplicações de análise de imagem, a segmentação é a primeira, mais importante e mais desafiadora etapa do processamento. Uma solução universal para a segmentação de células, aplicável a diferentes tipos de células e imagens, ainda não foi descrita. Por conseguinte, novos algoritmos e abordagens de segmentação tem sido constantemente propostos e desenvolvidos.

No âmbito desta tese, um algoritmo otimizado de segmentação foi desenvolvido, apresentado, testado e aplicado a um conjunto de imagens de contraste de fase da *Escherichia coli*. A solução proposta, na maioria dos casos, provou ser mais precisa e menos demorada do que a abordagem atualmente implementada pelo *Laboratory of Biosystem Dynamics*.

**Palavras-chave:** Segmentação, *Escherichia coli*, Microscopia de Contraste de Fase, Processamento de Imagem.



# Abstract

Nowadays, one of the main problems in biology field is related with the nature and behavior of cell cycle regulation mechanisms.

The Laboratory of Biosystem Dynamics, established in Finland, rely on the work of an inter-disciplinary research group to take forward in vivo, single-cell measurements in order to support several investigations. One of the main goals of these researches is to test and propose new computational tools and stochastic models to describe and allow further analysis of some of the mechanisms that regulate the cell cycle. Due to its relative simplicity, easy growth and manipulation, *Escherichia coli* becomes the ideal system to study these sorts of phenomena. Through phase contrast microscopy, a technique widely applied in biological and medical field, is possible to observe, with high-contrast, transparent organisms in their natural state.

In several microbiology labs, images are still often examined only by human visual inspection, which can be extremely subjective, time-consuming and lead to incorrect conclusions. Therefore, computational image processing and analysis are determinative to obtain statistically significant and reliable information. In many image analysis applications, segmentation is often the first, most important and most difficult processing stage. A universal solution to cell segmentation, applicable across cell and images types, has yet to be described. Therefore, new segmentation algorithms and approaches are continuously developed and required.

With this thesis, an improved segmentation algorithm was developed, presented, tested and applied on a set of *Escherichia coli* phase contrast images. The proposed solution, in most cases, proved to be more accurate and less time consuming than the one currently implemented by the Laboratory of Biosystem Dynamics.

**Keywords:** Segmentation, *Escherichia coli*, Phase Contrast Microscopy, Image Processing.



# Índice

<b>ÍNDICE DE FIGURAS</b>	<b>XV</b>
<b>ÍNDICE DE TABELAS</b>	<b>XIX</b>
<b>LISTA DE ABREVIATURAS E SIGLAS</b>	<b>XXI</b>
<b>1 INTRODUÇÃO</b>	<b>1</b>
1.2 <i>ESCHERICHIA COLI</i>	2
1.3 MICROSCOPIA DE CONTRASTE DE FASE	5
1.4 SEGMENTAÇÃO	7
1.4.1 <i>Breve resumo da história da análise celular</i> .....	8
1.4.2 <i>Abordagens mais comuns aplicadas na segmentação de células</i> .....	9
1.4.2.1 <i>Limiarização de Intensidade (Thresholding)</i> .....	11
1.4.2.2 <i>Extração de Características</i> .....	11
1.4.2.3 <i>Filtragem Morfológica</i> .....	13
1.4.2.4 <i>Crescimento de Regiões por Acumulação de Píxeis</i> .....	13
1.4.2.5 <i>Curvas Deformáveis</i> .....	14
1.4.3 <i>Análise estatística</i> .....	15
1.5 OBJETIVOS DA DISSERTAÇÃO	17
<b>2 FUNDAMENTAÇÃO TEÓRICA</b>	<b>19</b>
2.1 REPRESENTAÇÃO DA IMAGEM DIGITAL	19
2.2 O PÍXEL	20
2.3 PROCESSAMENTO DE IMAGEM	22
2.3.1 <i>Análise</i> .....	23
2.3.2 <i>Enriquecimento de Imagens</i> .....	23
2.3.3 <i>Etapas do Processamento de Imagem</i> .....	23
2.4 AQUISIÇÃO E DIGITALIZAÇÃO DE IMAGENS DIGITAIS	24
2.5 TÉCNICAS DE PRÉ-PROCESSAMENTO	27
2.5.1 <i>Histograma</i> .....	27
2.6 OPERAÇÕES PONTUAIS, LOCAIS E GLOBAIS	31
2.6.1 <i>Operação Pontual</i> .....	31
2.6.2 <i>Operação Local</i> .....	32

2.6.3 Operação Global.....	33
2.7 TRANSFORMAÇÕES DE INTENSIDADE	34
2.7.1 Função Identidade.....	34
2.7.2 Transformação Negativa .....	35
2.7.3 Ajuste de Contraste .....	36
2.7.4 Ajuste de Brilho .....	40
2.7.5 Ajuste Gama .....	41
2.8 SEGMENTAÇÃO	42
<b>3 METODOLOGIA ADOTADA</b>	<b>43</b>
3.1 MATLAB	43
3.2 INTERFACE GRÁFICA DO UTILIZADOR	44
3.2.1 Abrir e guardar ficheiros .....	45
3.2.2 Visualização e processamento de imagens.....	46
3.2.3 Análise e Segmentação .....	54
3.3 ALGORITMO	55
3.3.1 Imagens.....	57
3.3.2 Etapa 1: Filtragem.....	58
3.3.3 Etapa 2: Extração de contornos .....	59
3.3.4 Etapa 3: Seleção de bactérias.....	62
3.3.5 Etapa 4: Ordenação das bactérias selecionadas.....	63
3.3.6 Etapa 5: Criação do círculo central .....	65
3.3.7 Etapa 6: Setor circular direcional.....	72
3.3.8 Etapa 7: Propagação.....	75
<b>4 ANÁLISE DE RESULTADOS</b>	<b>79</b>
4.1 RESULTADOS OBTIDOS	79
4.2 VALIDAÇÃO DO ALGORITMO	80
4.3 ANÁLISE DE RESULTADOS	86
<b>5 CONCLUSÃO</b>	<b>89</b>
5.1 CONSIDERAÇÕES FINAIS	89
5.2 DESENVOLVIMENTOS FUTUROS	91
<b>BIBLIOGRAFIA</b>	<b>93</b>

# Índice de Figuras

<i>Figura 1.1: Diversidade das imagens celulares (retirado de [23]).....</i>	<i>10</i>
<i>Figura 1.2: Exemplos de segmentações de imagens de células (retirado de [23]).....</i>	<i>12</i>
<i>Figura 1.3: Análise estatística da literatura publicada sobre segmentação de imagens de células (adaptado de [23]).....</i>	<i>16</i>
<i>Figura 2.1: Imagem digital monocromática “Goldhill” com destaque para uma região da imagem de <math>17 \times 17</math> píxeis (retirado de [39]).....</i>	<i>20</i>
<i>Figura 2.2: Tipos de conectividade de píxeis (adaptado de [40]).....</i>	<i>21</i>
<i>Figura 2.3: Sistema de cores RGB (retirado de [42]).....</i>	<i>22</i>
<i>Figura 2.4: Etapas de um sistema de processamento de imagens (adaptado de [39]).....</i>	<i>24</i>
<i>Figura 2.5: Diferentes tipos de sensores de imagem (adaptado de [34]).....</i>	<i>25</i>
<i>Figura 2.6: (a) Imagem contínua projetada numa matriz de sensores de imagem. (b) Resultado da aplicação dos processos de quantização e amostragem à imagem. (retirado de [34]).....</i>	<i>26</i>
<i>Figura 2.7: Formação do vetor <math>p(x,y)</math> a partir dos valores dos píxeis correspondentes, presentes nos três componentes RGB que compõem a imagem (adaptado de [34]).....</i>	<i>27</i>
<i>Figura 2.8: Exemplos de histogramas (adaptado de [34]).....</i>	<i>29</i>
<i>Figura 2.9: Imagens correspondentes aos histogramas da figura 2.8 (retirado de [34]).....</i>	<i>30</i>
<i>Figura 2.10: Representação da aplicação de uma operação pontual (<math>T</math>) a um dado elemento <math>(m,n)</math> da imagem de entrada de modo a originar <math>g(m,n)</math>.....</i>	<i>31</i>
<i>Figura 2.11: Representação da aplicação de uma operação local (<math>T</math>) à imagem <math>f(m,n)</math> de modo a originar <math>g(m,n)</math>.....</i>	<i>33</i>
<i>Figura 2.12: Representação da aplicação de uma operação global (<math>T</math>) à imagem <math>f(m,n)</math> de modo a originar <math>g(m,n)</math>.....</i>	<i>33</i>
<i>Figura 2.13: Representação gráfica da aplicação de uma transformação de intensidade à imagem de entrada.....</i>	<i>34</i>
<i>Figura 2.14: Representação gráfica do mapeamento resultante da aplicação da função identidade a cada elemento da imagem original (retirado de [45]).....</i>	<i>35</i>
<i>Figura 2.15: Representação gráfica do mapeamento resultante da aplicação da transformação negativa a cada elemento da imagem original (adaptado de [45]).....</i>	<i>36</i>
<i>Figura 2.16: Resultado da aplicação da transformação negativa a uma imagem (retirado de [39]).....</i>	<i>36</i>
<i>Figura 2.17: Exemplo da aplicação do método de stretching ao histograma de uma imagem (adaptado de [46]).....</i>	<i>38</i>
<i>Figura 2.18: Representação gráfica do mapeamento resultante da aplicação da função sigmoide a cada elemento da imagem original (retirado de [45]).....</i>	<i>38</i>

<i>Figura 2.19: Resultado da aplicação da função sigmóide no contraste de uma imagem (retirado de [45]).....</i>	<i>39</i>
<i>Figura 2.20: Representação gráfica do mapeamento que caracteriza o aumento do brilho (para <math>k = 50</math>) de uma imagem (retirado de [45]).....</i>	<i>41</i>
<i>Figura 2.21: Representação gráfica do mapeamento que caracteriza a correção gamma (para <math>\gamma = [1.0;1.8;2.0;2.2;2.5]</math>) de uma imagem (retirado de [47]).....</i>	<i>42</i>
<i>Figura 3.1: Interface gráfica do utilizador desenvolvida no âmbito desta tese .....</i>	<i>45</i>
<i>Figura 3.2: Diferentes tipos de mapeamento associados ao ajuste do gamma da imagem original através da aplicação da função imadjust (retirado de [55]).....</i>	<i>47</i>
<i>Figura 3.3: Mapeamento correspondente a um aumento do contraste da imagem original (adaptado de [56]). .....</i>	<i>47</i>
<i>Figura 3.4: Mapeamento correspondente a uma diminuição do brilho da imagem original (retirado de [56]).....</i>	<i>48</i>
<i>Figura 3.5: Filtro de média representado por uma máscara do tipo <math>3 \times 3</math> (adaptado de [57])...50</i>	
<i>Figura 3.6: Representação gráfica de como computar a mediana de uma dada região da imagem (adaptado de [58]).....</i>	<i>51</i>
<i>Figura 3.7: Exemplo de uma máscara de convolução (<math>5 \times 5</math> píxeis) para a aplicação do filtro Gaussiano (com <math>\sigma = 1.4</math>) (retirada [59]).....</i>	<i>52</i>
<i>Figura 3.8: Três exemplos comuns de máscaras de convolução (<math>3 \times 3</math>) para a aplicação do filtro Laplaciano (retirada de [60]). .....</i>	<i>52</i>
<i>Figura 3.9: Máscaras (<math>3 \times 3</math>) utilizadas para a aplicação (vertical e horizontal) do filtro Sobel (adaptado de [61]).....</i>	<i>53</i>
<i>Figura 3.10: Representação gráfica do procedimento associado ao algoritmo de segmentação proposto .....</i>	<i>56</i>
<i>Figura 3.11: Exemplo das imagens disponibilizadas pelo Laboratory of Biosystem Dynamics. A imagem destacada na figura representa apenas uma porção, composta por <math>681 \times 601</math> píxeis, da imagem original.....</i>	<i>58</i>
<i>Figura 3.12: Exemplo da aplicação do filtro Gaussiano para a atenuação do ruído da imagem .....</i>	<i>59</i>
<i>Figura 3.13: Histograma e histograma acumulado da imagem presente na Figura 3.11.....</i>	<i>61</i>
<i>Figura 3.14: Exemplo da aplicação do processo de binarização baseado no método de Otsu à imagem suavizada.....</i>	<i>61</i>
<i>Figura 3.15: Exemplo da aplicação do processo do operador Laplaciano à imagem binarizada .....</i>	<i>62</i>
<i>Figura 3.16: Exemplo do resultado obtido através da etapa da seleção de bactérias.....</i>	<i>63</i>
<i>Figura 3.17: Aproximação da morfologia da E. coli a um modelo geométrico constituído por um retângulo e dois semicírculos .....</i>	<i>64</i>
<i>Figura 3.18: Tipos de vizinhança detetados na etapa da ordenação das bactérias selecionadas .....</i>	<i>65</i>



<i>Figura 3.19: Anel circular que representa a vizinhança considerada para cada bactéria selecionada na etapa da criação do círculo central.....</i>	<i>66</i>
<i>Figura 3.20: (A) Interseção da circunferência de vizinhança com uma das bordas da bactéria; (B) interseção da circunferência de vizinhança com ambas as laterais da bactéria considerada.....</i>	<i>67</i>
<i>Figura 3.21: Representação do vetor que une o centro da bactéria considerada ao primeiro píxel branco a ser detetado.....</i>	<i>68</i>
<i>Figura 3.22: Resultados exemplificativos da aplicação da etapa da criação do círculo central em bactérias inseridas numa vizinhança inicial do tipo 1.....</i>	<i>68</i>
<i>Figura 3.23: Representação do tipo de ajuste aplicado sempre que são detetadas duas bactérias a uma distância igual ou superior a <math>2 \times r - 1</math> píxeis do centro da bactéria considerada.....</i>	<i>69</i>
<i>Figura 3.24: Representação do vetor que une o centro da bactéria considerada ao primeiro píxel colorido a ser detetado.....</i>	<i>70</i>
<i>Figura 3.25: Resultados exemplificativos da aplicação da etapa da criação do círculo central em bactérias inseridas numa vizinhança inicial do tipo 2.....</i>	<i>70</i>
<i>Figura 3.26: Resultados exemplificativos da aplicação da etapa da criação do círculo central em bactérias inseridas numa vizinhança inicial do tipo 3.....</i>	<i>71</i>
<i>Figura 3.27: Representação gráfica do tipo de ajuste a aplicar a bactérias que, na etapa da criação do círculo central, se inseriram numa vizinhança do tipo 4.....</i>	<i>72</i>
<i>Figura 3.28: Setor circular de raio <math>R</math> e ângulo de abertura <math>\theta</math> (adaptado de [65])......</i>	<i>72</i>
<i>Figura 3.29: Ilustração do processo a partir do qual se forma o par de setores circulares. O centro do sistema de eixos da figura representa o ponto da imagem definido como o centro de uma dada bactéria considerada. O raio do semicírculo que se forma entre os 0 e os <math>2\pi</math> radianos (relativamente ao eixo das abcissas da imagem) é igual a <math>R</math>. <math>[0;2\pi]</math> representa o intervalo de ângulos para o qual será testada a hipótese da formação de cada reta que poderá compor o par de setores circulares (adaptado de [66])......</i>	<i>74</i>
<i>Figura 3.30: Exemplo do resultado da aplicação da etapa da formação dos setores circulares direcionais.....</i>	<i>75</i>
<i>Figura 3.31: Restrição do intervalo de ângulos para o qual será testada a hipótese da formação de cada reta que poderá compor o próximo par de setores circulares direcionais (adaptado de [66])......</i>	<i>76</i>
<i>Figura 3.32: Exemplo do resultado da aplicação da etapa da propagação de círculos.....</i>	<i>77</i>
<i>Figura 4.1: Exemplos do resultado da aplicação do algoritmo de segmentação proposto a diversos conjuntos de bactérias.....</i>	<i>79</i>
<i>Figura 4.2: (A) Imagem original; (B) Contornos das bactérias presentes na imagem original extraídos a partir do método apresentado na secção 3.3.3; (C) Sobreposição da imagem original com os contornos extraídos; (D) Sobreposição da segmentação manual com os contornos extraídos.....</i>	<i>84</i>
<i>Figura 4.3: Comparação da sensibilidade obtida para o algoritmo proposto considerando diferentes tipos de aproximações entre a morfologia dos contornos obtidos pelo algoritmo</i>	

<i>desenvolvido e a morfologia dos contornos definida pelos resultados da segmentação manual</i>	85
<i>Figura 4.4: Comparação da sensibilidade obtida para o algoritmo proposto considerando diferentes tipos de aproximações entre a morfologia dos contornos obtidos pelo algoritmo desenvolvido e a morfologia dos contornos definida pelos resultados da segmentação manual</i>	85
<i>Figura 4.5: Comparação entre alguns dos resultados de segmentação obtidos pelo algoritmo automático desenvolvido pelo LBD (para cada exemplo, à esquerda) e os resultados obtidos através da aplicação do algoritmo de segmentação proposto no âmbito desta tese (para cada exemplo, à direita)</i>	87

# Índice de Tabelas

<i><b>Tabela 4.1:</b> Parâmetros de validação do algoritmo de segmentação proposto calculados para 50 imagens. ....</i>	<i><b>82</b></i>
<i><b>Tabela 4.2:</b> Resumo dos valores dos parâmetros de validação do algoritmo de segmentação proposto obtidos para cada grupo de imagens avaliado. ....</i>	<i><b>83</b></i>



# Lista de abreviaturas e siglas

**ADN ou DNA** *Ácido Desoxirribonucleico*

**CCD** *Charge Coupled Device*

***E. coli*** *Escherichia coli*

**GUI** *Graphical User Interface*

**GUIDE** *Graphical User Interface Development Environment*

**IPT** *Image Processing Toolbox*

**LBD** *Laboratory of Biosystem Dynamics*

**LOG** *Laplacian of Gaussian*

**MATLAB** *Matrix Laboratory*

**qPCR** *quantitative Polymerase Chain Reaction*





# Introdução

Um dos maiores desafios associados à área da biologia relaciona-se com a compreensão do comportamento e natureza dos mecanismos que regulam o ciclo celular. Nos dias de hoje, a maioria das investigações a respeito deste ciclo incidem essencialmente sobre os processos que regulam o tempo e a frequência de fenómenos tão fulcrais como a replicação do DNA (ácido desoxirribonucleico), segregação e divisão celular [1].

Anteriormente, acreditava-se que as populações de células eram homogêneas. No entanto, evidências confirmam que, mesmo considerando apenas pequenas populações, a heterogeneidade existe. Assim sendo, análises e medições que se baseiem no princípio da existência de populações homogêneas conduzem a resultados enganosos por não serem consideradas as pequenas e cruciais diferenças existentes entre células individuais - as quais podem variar fortemente tanto do ponto de vista morfológico como fisiológico. Acredita-se que nestas ínfimas variações possa residir a chave para melhor compreender e responder a inúmeras questões nunca antes esclarecidas associadas a áreas de investigação tão eminentes como o cancro, a imunologia e a neurologia [2].

O *Laboratory of Biosystem Dynamics* (LBD), estabelecido na Finlândia, é constituído por um grupo de investigação interdisciplinar que, na tentativa de melhor compreender a célula e os seus mecanismos, realiza múltiplos estudos e medições, *in vivo*, os quais, maioritariamente, são aplicados ao nível celular (em inglês, *single-cell analysis*) ou inclusivamente ao nível molecular (em inglês, *single-molecule analysis*) [3]. Este tipo de análise encarrega-se do estudo do genoma, proteínas, transcritos e metabolismos tendo sempre em consideração a individualidade de cada célula [4].

Para a realização destes estudos, o LBD possui ao seu dispor um conjunto abrangente de equipamentos e técnicas derivados da Biologia Molecular, tais como a espectrofotometria de microplacas, qPCR (do inglês, *quantitative polymerase chain reaction*) e *Western blotting*. A aplicação destes procedimentos é frequentemente combinada com uma forte componente de observação microscópica em regime *time-lapse* [3].

A microscopia *time-lapse* é uma técnica de imagiologia que permite, em intervalos de tempo predeterminados e regulares, capturar um conjunto de imagens respeitantes a um determinado evento dinâmico. Através desta técnica é possível observar e analisar fenómenos tão específicos tais como a mitose, o nível de oxigénio de uma dada célula ou até mesmo o movimento das proteínas e lípidos através membrana celular ao longo do tempo [5]. A microscopia *time-lapse*, quando aplicada ao estudo de células vivas, pode ser comprometida por um conjunto de fatores diversificados que inclui, entre outros, flutuações na temperatura, pH, humidade e iluminação. Perante tais circunstâncias, para além da capacidade de foco do aparelho ser diretamente afetada, podem inclusivamente surgir danos irreversíveis nas células da amostra [6].

Com o auxílio da microscopia de *time-lapse* e de outras técnicas laboratoriais tem sido possível ao LBD, ao longo do tempo, o estudo de fenómenos tão inexplorados tais como as dinâmicas da transcrição, a segregação de agregados indesejados, o envelhecimento celular e alguns dos mecanismos responsáveis pela regulação da expressão génica nas bactérias [3].

Um dos principais objetivos deste grupo é compreender, de uma forma mais pormenorizada, o modo como os genes e as dinâmicas envolvidas no circuito genético são regulados, de modo a poder desenvolver e propor novos modelos estocásticos e ferramentas computacionais capazes de descrever correta e detalhadamente este tipo de fenómenos promovendo, simultaneamente, um método de análise mais aprofundada dos mesmos [3].

A maioria das investigações desenvolvidas pelo LBD baseia-se em observações e análises detalhadas do comportamento da *Escherichia coli* (*E. coli*). A descrição deste microrganismo é apresentada, de um modo sucinto, no tópico seguinte.

## 1.2 *Escherichia coli*

Todos os anos, milhares de pessoas adoecem e centenas delas morrem devido a doenças provocadas pela ação da *Escherichia coli* [7].



*Escherichia coli*, originalmente denominada por “*Bacterium coli commune*”, é uma bactéria bacilar Gram-negativa que pertence à família *Enterobacteriaceae* e que habita, normalmente, no trato gastrointestinal do ser humano e de outros animais de sangue quente [8].

O termo “bacilar” é usado para caracterizar bactérias cuja morfologia se aproxima à de um bastonete. Esta geometria é definida pela parede celular do microrganismo, uma estrutura complexa, rígida e constituída por diversas camadas. Para além deste tipo de categorização, as bactérias, podem ainda ser classificadas enquanto Gram-positivas ou Gram-negativas, consoante apresentem uma coloração azul ou vermelha – respetivamente - após a aplicação de um corante, denominado corante de Gram [9].

A maioria das estirpes da *Escherichia coli* podem ser consideradas inofensivas e até mesmo contribuir com alguns benefícios para a saúde dos seus hospedeiros. Dois exemplos destas contribuições são a produção de vitamina K2 - a qual desempenha um papel fundamental na coagulação do sangue, prevenção de doenças cardíacas e manutenção da saúde óssea - e a supressão do desenvolvimento de algumas bactérias patogénicas que possam proliferar no trato gastrointestinal [4,10]. Dentro da população da *E. coli*, no entanto, residem quatro estirpes consideradas potencialmente perigosas por serem capazes de provocar um conjunto de patologias – tais como diarreia, febre, dores musculares e de cabeça, infeções do trato urinário, gastroenterite e meningite – as quais, podem afetar severamente a saúde do hospedeiro. Estas doenças, normalmente, são desencadeadas por um conjunto de mecanismos específicos desta bactéria que conduzem à produção de toxinas, perturbações no metabolismo celular e destruição de tecidos [11].

De um ponto de vista bioquímico, fisiológico e genético, a *E. coli* é um dos organismos vivos melhor compreendidos e caracterizados [1]. No que diz respeito à sua fisionomia, esta bactéria, anaeróbica facultativa (assim caracterizada por possuir a capacidade de se reproduzir e desenvolver tanto em ambientes ricos como em ambientes em total ausência de oxigénio) com aproximadamente 0.5-1.0 µm de diâmetro e 1.0-3.0 µm de comprimento, pode ainda possuir múltiplos flagelos - os quais se dispõem em redor da mesma e permitem com que esta se consiga movimentar facilmente em meios líquidos [8].

A respeito das condições de temperatura e pH favoráveis ao crescimento da *Escherichia coli*, pode concluir-se, de um modo geral, que a maioria das estirpes desta bactéria consegue proliferar em ambientes cuja temperatura se encontre entre os 15 e os 48 °C, sendo que a temperatura ótima para este processo (temperatura à qual a taxa de crescimento é máxima) se encontra compreendida na gama dos 37-42°C. Apesar do valor de pH ótimo ao crescimento da *E. coli* ser um valor de pH neutro (i.e., 7.0), estudos indicam que este microrganismo é capaz de se desenvolver normalmente em meios cujo pH esteja entre 5.5 e 8.0. [8]

Quanto ao processo de reprodução bacteriana (o qual ocorre por intermédio da fissão binária), podem ser destacadas três etapas principais que conduzem à formação de duas “células-filhas”, geneticamente idênticas, a partir de uma única “célula-mãe”: alongamento celular, replicação do DNA e divisão do citoplasma. O tempo necessário para uma célula se dividir (e a sua população duplicar) é denominado por tempo de geração. Este parâmetro varia consideravelmente entre organismos e depende significativamente de uma série de condições ambientais, tais como a temperatura e o pH do meio envolvente [12].

Devido à sua simplicidade, as células procarióticas (bactérias) têm servido de objeto para o estudo de inúmeros mecanismos celulares e de outros aspetos inerentes às áreas da bioquímica, biologia e genética molecular. A espécie bacteriana mais amplamente abrangida por estes estudos, dada a facilidade com que pode ser reproduzida, manipulada e observada em laboratório, é a *E. coli*. Recorrendo a inúmeras investigações baseadas neste microrganismo, foi possível atingir-se a compreensão de conceitos tão relevantes como o código genético, expressão génica, síntese proteica e replicação do DNA [13]. Neste conjunto de investigações pode incluir-se, a título de exemplo, muitos dos estudos desenvolvidos pelo LBD.

Sabe-se atualmente que o genoma da *E. coli* (totalmente sequenciado em 1997) é constituído aproximadamente por 4.6 milhões de pares de bases e cerca de 4000 genes. Considerando o genoma humano, o qual é praticamente mil vezes maior, a dimensão reduzida do genoma desta bactéria representa, ainda hoje, uma enorme vantagem para estudos e análises principalmente relacionados com a área da genética. Para além da vantagem anteriormente referida, este tipo de experimentos é facilitado também pela rápida proliferação da *E. coli* a qual, em condições ótimas de cultura, se divide a cada 20 minutos. Assim sendo, após 20 gerações, uma única célula-mãe inicial terá originado, em apenas cerca de 7 horas, uma descendência equivalente a mais de um milhão de células [14].

Ao longo do tempo, uma grande parte dos estudos óticos baseados na observação e análise da *Escherichia coli* têm sido condicionados pelo reduzido tamanho deste microrganismo – o qual não é muito maior do que o limite de resolução da maioria dos microscópios óticos,  $\sim 0.25 \mu\text{m}$  [1]. A resolução de um sistema ótico quantifica a sua capacidade de distinguir individualmente objetos adjacentes de uma imagem. O limite de resolução ( $\delta$ ) é definido enquanto a menor distância entre dois pontos capaz de ser distinguida pelo sistema ótico. Quanto menor for o limite de resolução do dispositivo, melhor será a definição da imagem. O poder de resolução é dado pelo inverso desta medida [15].

Atualmente, a limitação destacada no parágrafo anterior pode ser contornada através de uma análise cuidadosa de imagens geradas por microscopia de fluorescência e contraste de fase

em regime *time-lapse*. Por intermédio da microscopia de fluorescência, o nucleóide bem como algumas das restantes estruturas e organelos celulares, podem ser facilmente identificados, observados e analisados. Assim sendo, esta técnica traduz-se numa ferramenta através da qual investigadores conseguem aceder uma ampla gama de informações do tipo funcional. Por sua vez, através da microscopia de contraste de fase, é possível extrair-se um conjunto de informações morfológicas de elevada relevância científica. A sobreposição destes dois tipos de imagens gera assim uma série de informações novas e cruciais que permitem melhor compreender e analisar inúmeros processos biológicos [16].

De seguida, irá descrever-se, mais pormenorizadamente, a técnica de microscopia por contraste de fase por se tratar do método através do qual foram obtidas as imagens que constituíram o objeto de estudo da presente dissertação.

## 1.3 Microscopia de contraste de fase

Os microscópios são instrumentos que possuem como principal objetivo o de gerar imagens aumentadas de objetos, maioritariamente, impossíveis de serem visualizados e analisados a olho nu. Para além da sua notável capacidade de ampliação, os microscópios contribuem ainda com uma segunda grande potencialidade: o elevado poder de resolução associado às imagens geradas. Quanto maior for esta capacidade, maior será a definição da imagem adquirida [17].

Atualmente, estes aparelhos dividem-se em duas categorias principais: os microscópios óticos e os microscópios eletrónicos. No caso do microscópio ótico, o qual se pensa ter sido criado por Hans Janssen e pelo seu filho Zacharias Janssen entre os anos de 1590 e 1595, a aquisição de imagens, ampliadas através de um sistema de lentes combinadas, baseia-se em fenómenos físicos resultantes da interação de um feixe de luz monocromático e coerente com a amostra e o seu meio envolvente. O poder de resolução associado a estes aparelhos é aproximadamente igual a 0.25  $\mu\text{m}$ , o que corresponde a cerca de mil vezes mais do que o poder de resolução do olho humano [18].

A microscopia de contraste de fase, uma variante da microscopia ótica, é uma técnica largamente empregue para produzir imagens altamente contrastadas de espécimes transparentes (como é o caso das células vivas, microrganismos, fibras, dispersões de latex e partículas subcelulares - incluindo o núcleo e outros organelos) [19]. O microscópio de contraste de fase, uma invenção do físico alemão Frits Zernike e que lhe valeu um prémio Nobel em 1953, distingue-se assim dos restantes dispositivos de microscopia por possuir um sistema ótico que possibilita a

observação de estruturas transparentes que difiram ligeiramente entre si ao nível do seu índice de refração [18].

O índice de refração ( $n$ ) é uma grandeza adimensional que correlaciona a velocidade da luz no vácuo ( $c$ ) com a velocidade da luz num determinado meio material ( $v$ ) [20]. Quanto menor for o índice de refração do meio sobre o qual a luz incide (isto é, quanto mais próximo o valor do desse índice de refração estiver de 1) maior será a velocidade com que a luz trespassará o mesmo.

Esta relação pode ser descrita pela fórmula abaixo apresentada na qual  $c$  representa a velocidade da luz no vácuo ( $3 \times 10^8$  m/s) e  $v$  a velocidade com que a luz atravessa o meio em análise.

$$n = \frac{c}{v}$$

Considerando dois meios transparentes, A e B, e um feixe de luz orientado de A para B, para que haja refração (isto é, mudança na direção de propagação da onda ao atravessar a fronteira entre os dois meios) é necessário que  $n_A \neq n_B$ .

Na microscopia de contraste de fase, sempre que o índice de refração do espécime for maior do que o do meio envolvente, ao atravessar a amostra, a onda luminosa incidente diminui em velocidade e é, posteriormente, retardada em fase. Por outro lado, sempre que o índice de refração do meio envolvente exceder o do espécime, a velocidade de propagação da onda luminosa que atravessa o mesmo aumenta e, seguidamente, é acelerada em fase [19]. A diferença de fase imposta à porção da onda incidente que atravessa o espécime pode definida, em radianos, por:

$$\delta = 2\Delta\pi/\lambda$$

Na equação acima destacada,  $\lambda$  corresponde ao comprimento de onda da radiação visível incidente e o termo  $\Delta$  representa a diferença de percurso ótico, a qual é definida por:

$$\Delta = (n_2 - n_1) \times t$$

onde  $n_2$  constitui o índice de refração do espécime,  $t$  a sua espessura e  $n_1$  o índice de refração do meio envolvente.

Sempre que um objeto transparente, com um índice de refração próximo ao do meio envolvente, é atravessado por um raio luminoso, ocorre uma divisão na qual uma parte desse raio contorna ou atravessa a amostra sem sofrer qualquer tipo de interação com a mesma (onda S, *surround wave*), enquanto a outra (onda D, *D-wave*) interage com o espécime sendo difratada pelo mesmo. Posteriormente, estas duas componentes acabam por se combinar produzindo uma onda resultante denominada por P (*particle wave*) [19]. A relação matemática que se estabelece

entre as diferentes ondas luminosas geradas por intermédio desta interação pode ser descrita, de um modo muito simples, pela seguinte expressão:

$$P = S + D$$

Na microscopia de contraste de fase, a detecção do espécime na imagem depende da diferença relativa entre as intensidades (ou amplitudes) da onda S e P. Caso as amplitudes da onda S e da onda P sejam significativamente diferentes, o espécime adquire uma quantidade significativa de contraste e é facilmente observado. De outro modo, o espécime permanece transparente.

Ao atravessarem áreas definidas por diferentes comprimentos de percurso ótico,

$$\text{comprimento de percurso ótico} = n \times t$$

os raios luminosos podem ser retardados até  $\frac{1}{4}\lambda$ , no entanto, a sua amplitude permanece constante. Uma vez que o olho humano não consegue discernir diferenças de fase, é necessário um sistema ótico capaz de transformar essas variações em diferenças de amplitude.

O princípio do funcionamento do microscópio de contraste de fase baseia-se então na conversão de pequenas variações de fase em diferenças de intensidade, isto é, zonas claras e escuras, as quais podem ser percecionadas enquanto acentuações no contraste da imagem

Este tipo de conversão pode resultar numa imagem de contraste de fase positiva, sempre que o espécime em análise apareça mais escuro do que o plano de fundo, ou numa imagem de contraste de fase negativa, sempre que o espécime em análise apareça mais claro do que o plano de fundo.

Uma das maiores vantagens associadas a esta técnica de microscopia consiste no facto das células poderem ser observadas e analisadas no seu estado natural isto é, vivas e sem necessitarem de ser fixadas e expostas a corantes [19].

## 1.4 Segmentação

Desde que se estabeleceu a teoria celular no início do século XIX, a qual reconhece a célula enquanto a unidade básica da vida, biólogos têm procurado explicar os mais diversos fenómenos e princípios subjacentes a esta estrutura. Inúmeras descobertas nesta área foram alcançadas ao longo de muitas décadas de investigação [21]. No entanto, a busca pela compreensão plena dos mecanismos celulares e a melhor maneira de manipulá-los a favor da saúde permanece até aos dias de hoje, envolvendo um orçamento e número de pessoas cada vez maior e um conjunto de ferramentas mais sofisticadas do que nunca [22].

Atualmente, biólogos e investigadores não só possuem à sua disposição uma infinidade de técnicas distintas de imagem microscópica como também usufruem de um conjunto de sistemas avançados que lhes permitem adquirir, num intervalo de apenas algumas horas, um número muito elevado de imagens. Apesar destes progressos tecnológicos, em muitos dos laboratórios de microscopia, as imagens adquiridas continuam a ser analisadas exclusivamente através da inspeção visual, o que pode tornar o processo bastante subjetivo, demorado e conduzir a conclusões incorretas. Assim sendo, a análise computacional de imagens é determinante para a obtenção de informações fidedignas e estatisticamente significantes.

### **1.4.1 Breve resumo da história da análise celular**

Foi há mais de meio século atrás que, pela primeira vez, um computador foi utilizado com o propósito de analisar imagens celulares [23].

Em meados da década de 1950, surgem os primeiros programas computacionais capazes de, automaticamente, classificar um conjunto de esfregaços (lâminas de vidro sobre as quais são colocadas leves camadas de matéria orgânica para análise microscópica) com o objetivo de se fazer uma triagem em massa relativa ao cancro cervical. Inicialmente, estes algoritmos eram aplicados unidimensionalmente, baseando as suas decisões na comparação de um limiar de intensidade, previamente definido, com o valor de cada píxel pertencente a cada linha (1D) da imagem microscópica [24].

Em 1960 surgem os primeiros exemplos de processamento automático de imagens a duas dimensões (2D) os quais, baseando-se em medidas morfológicas e colorimétricas bastante simples, foram desenvolvidos com o intuito de se efetuar uma contagem diferencial de glóbulos brancos (leucócitos) [25]. Este tipo de sistemas começaram a ser comercializados em meados da década de 1970, passando a incluir múltiplos circuitos elétricos que possibilitavam a execução simultânea de diferentes tarefas de observação e análise. Foi também nesta altura que os primeiros microscópios assistidos por computador foram desenvolvidos para a análise morfológica de células neuronais [26].

Com a chegada da microscopia confocal (técnica microscópica que permite, através do agrupamento de um conjunto de cortes óticos, reconstruir a topografia de objetos complexos), na década de 1980, novas portas foram abertas para análise tridimensional (3D) de imagens de células. No entanto, foi apenas na década de 1990, quando os computadores se tornaram poderosos o suficiente para lidar com dados a três dimensões, ou até mesmo dados mais complexos a duas

dimensões, que as comunidades do processamento de imagem e visão computacional começaram a encarar este desafio [27].

Desde então, a literatura publicada sobre esta temática tem crescido de um modo exponencial (ver **Figura 1.3**) e uma grande parte dos métodos de análise de imagem propostos aplicada às mais distintas áreas de estudos [23].

## 1.4.2 Abordagens mais comuns aplicadas na segmentação de células

Abundância, heterogeneidade e complexidade são três dos conceitos que melhor definem a gestão, o processamento e a análise manual de imagens microscópicas. Para suprir estas condições, inúmeras técnicas computacionais, desenhadas para executar este tipo de tarefas, têm sido desenvolvidas ao longo do tempo [28].

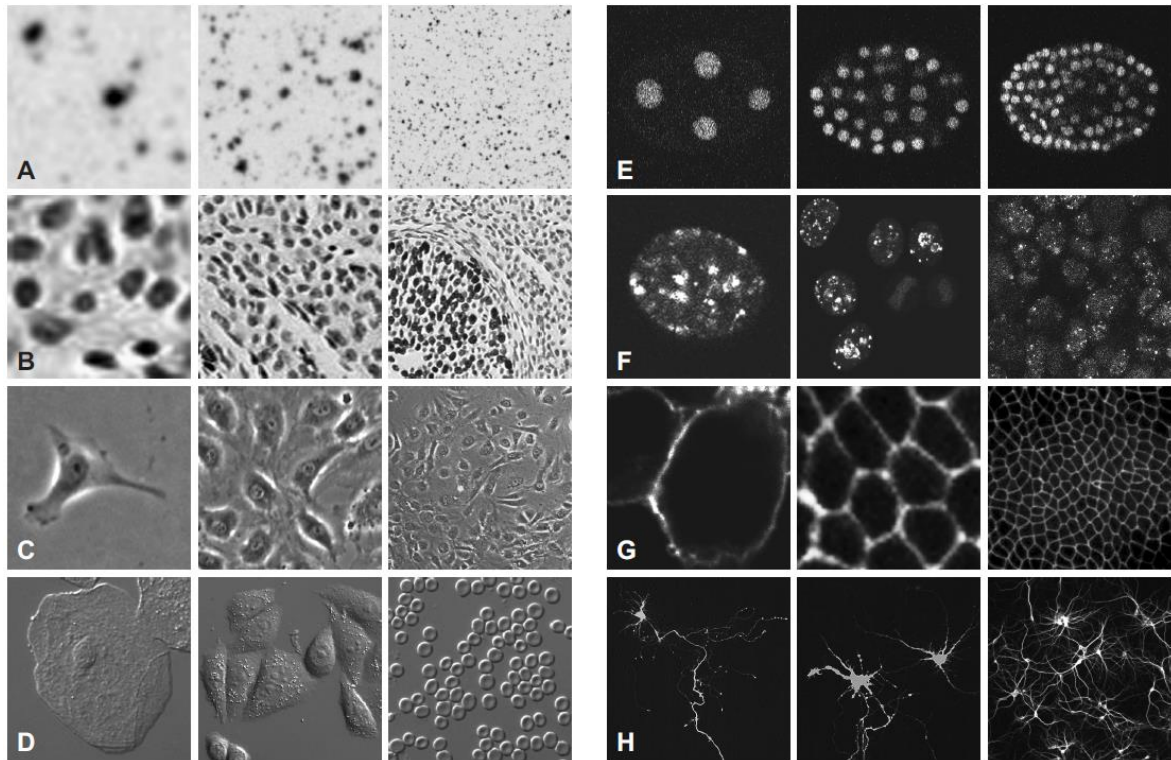
Na maioria das vezes, antes de se extrair qualquer tipo de informação a partir de uma imagem, é necessário pré processar-se os seus dados de modo a corrigir eventuais desequilíbrios na sua iluminação, reduzir o ruído e/ou realçar determinadas características como por exemplo os contornos e as bordas dos objetos presentes na imagem [29].

A segmentação, o próximo passo a ser aplicado, é o processo através do qual uma imagem é decomposta em partes distintas. Tipicamente, este procedimento é aplicado a imagens digitais com o objetivo de identificar diferentes objetos ou outro tipo de informações consideradas relevantes que sejam caracterizados por atributos similares [30].

Para que seja possível analisar e interpretar corretamente uma imagem, é necessário obter uma segmentação fidedigna da mesma. No entanto, segmentar uma imagem com um elevado grau de precisão traduz-se, normalmente, num problema bastante desafiador [31]. De um modo geral, algoritmos que funcionam bem num determinado conjunto de imagens produzem resultados de segmentação insuficientes quando aplicados a um outro conjunto com características diferentes. Para além disso, quando aplicada ao caso específico das imagens celulares, a elevada diversidade (variedade de técnicas microscópicas aplicadas, coloração, tipos de células e densidades celulares) e complexidade dos dados analisados contribuem para aumentar o grau de dificuldade desta tarefa (ver **Figura 1.1**). Uma solução universal para a segmentação de células, capaz de ser aplicada a diferentes tipos de células e imagens, ainda não foi descoberta [32].

Na maioria das aplicações desenvolvidas na área da análise de imagem, a segmentação constitui a primeira, mais importante e mais complexa etapa do processamento da imagem. Consequentemente, inúmeros algoritmos de segmentação tem sido propostos, ao longo das últimas décadas, e adaptados a um amplo conjunto de diferentes áreas de investigação. A análise de imagens médicas é um dos domínios que mais tem atraído a atenção de investigadores devido às potenciais contribuições na melhoria das condições da saúde humana que daí podem advir [22].

Ao analisar-se a literatura publicada sobre esta temática, constatou-se que a grande maioria dos métodos de segmentação propostos podem ser resumidos num conjunto bastante reduzido de abordagens. As mais comuns serão, em seguida, sucintamente apresentadas com base na informação discriminada em [23].



**Figura 1.1: Diversidade das imagens celulares.** As imagens de células podem variar bastante no que diz respeito à técnica de microscopia e coloração aplicadas bem como o tipo de células analisadas e a respetiva densidade celular. Nesta figura são apresentados diversos exemplos ilustrativos de imagens celulares (com uma densidade que aumenta da esquerda para a direita) adquiridas por microscopia ótica de campo claro (A-B), microscopia de contraste de fase (C), microscopia de interferência diferencial (D) e microscopia de fluorescência (E-H) (retirado de [23]).



### 1.4.2.1 Limiarização de Intensidade (*Thresholding*)

A limiarização de intensidade é uma técnica de segmentação, bastante simples e computacionalmente pouco pesada, que se baseia na diferença dos níveis de intensidade (ou níveis de cinzento) que constituem os diferentes objetos de uma imagem. Normalmente, os píxeis da imagem pertencentes às células possuem valores de intensidade significativamente diferentes dos píxeis que constituem o plano de fundo da imagem (*background*). Este foi o primeiro método de segmentação a ser desenvolvido e, ironicamente, nos dias de hoje, continua a ser a abordagem mais frequentemente adotada no processo da segmentação de células [33].

O método mais simples de limiarização consiste em substituir cada píxel da imagem original por um píxel de cor preta ou branca sempre que a intensidade da imagem for, respetivamente, inferior ou superior a um determinado limiar de intensidade previamente estabelecido. Este limiar (ou *threshold*) pode ser aplicado quer globalmente, definindo-se um único valor de intensidade (fixo) para toda a imagem, quer localmente, exigindo a aplicação de um outro tipo de limiar denominado por limiar dinâmico ou adaptativo. Geralmente, as abordagens automáticas desenvolvidas para determinar o melhor valor de limiar baseiam-se na análise estatística (global ou local) do histograma – uma representação gráfica da distribuição dos níveis de intensidade na imagem [34].

Por vezes, quando o *background* não se encontra suficientemente uniformizado ou quando a suposição inicial de que as células possuem um nível de intensidade consideravelmente diferente do nível do plano de fundo falha, o processo de limiarização de intensidade torna-se extremamente difícil e, por si só, na maioria das vezes, conduz a resultados de segmentação considerados insuficientes. Assim sendo, esta metodologia acaba por ser útil, na maioria dos casos, apenas enquanto primeira ou última etapa do processo de segmentação.

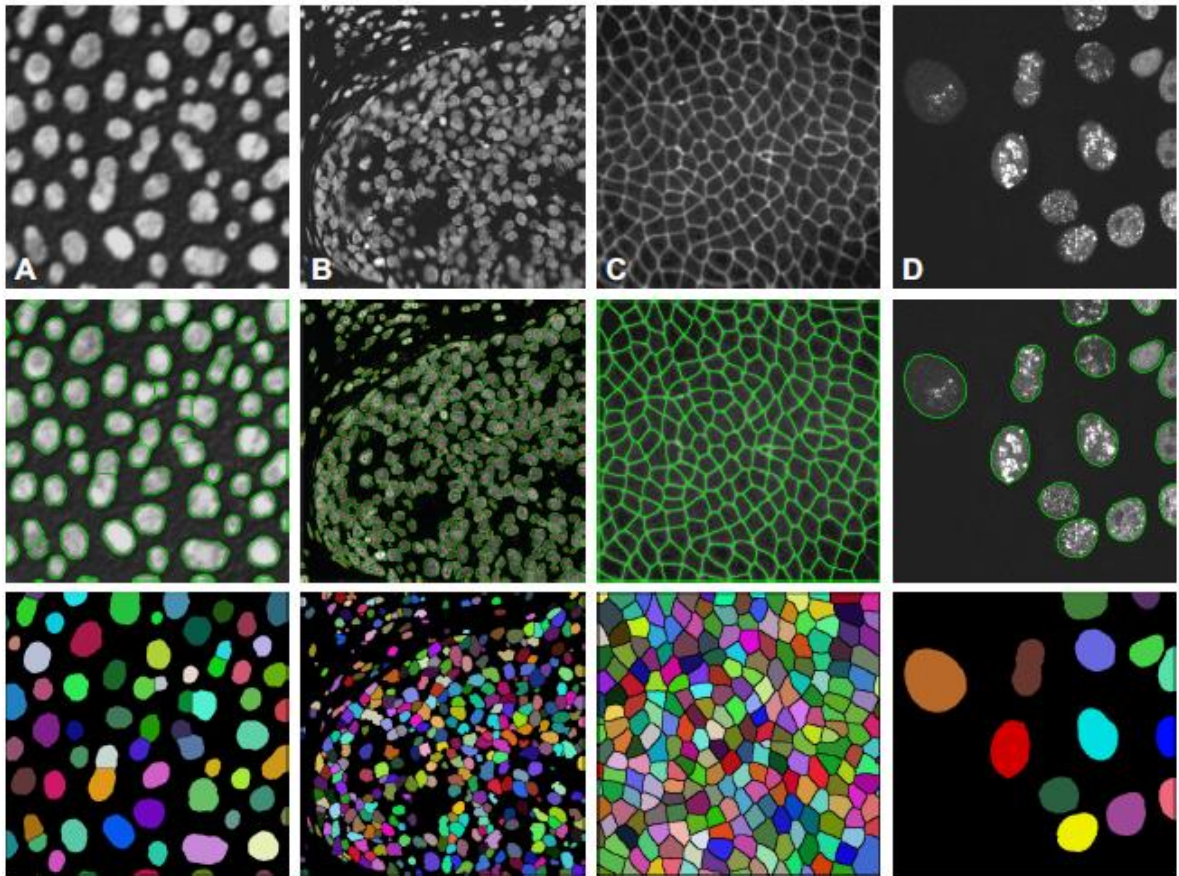
### 1.4.2.2 Extração de Características

Para além de poderem ser distinguidas com base nas suas intensidades absolutas, as células podem ainda ser segmentadas através da extração de características derivadas de intensidade (são exemplo de tais características, as bordas e os contornos dos objetos presentes na imagem). Estes atributos podem ser realçados mediante a aplicação de um conjunto de filtros lineares à imagem [23].

O processo de filtragem, abordado mais detalhadamente no tópico seguinte, pode ser resumidamente definido como uma operação de vizinhança capaz de modificar, remover ou realçar

determinadas características (ou atributos) da imagem. Os filtros lineares constituem uma categoria específica de filtros que atuam na imagem substituindo o valor de intensidade de um determinado píxel central por uma combinação linear dos valores dos píxeis da sua vizinhança [34].

Quando observadas através de um sistema de baixa ampliação, as células assemelham-se a partículas compactas que podem ser realçadas mediante a aplicação de um detetor de blob (detetor de região de interesse) como por exemplo, o filtro Laplaciano ou o filtro Laplaciano do Gaussiano (LOG) (ver **Figura 1.2**). Perante outro tipo de cenários, filtros diferenciais de primeira e segunda ordem, precedidos ou sucedidos por outro tipo de metodologias, são frequentemente aplicados. Semelhante ao que se verifica com o método de limiarização de intensidade, a aplicação deste conjunto de filtros, por si só, geralmente não produz bons resultados de segmentação [23].



**Figura 1.2: Exemplos de segmentações de imagens de células.** Nas três linhas que compõe a presente figura são destacados, respetivamente, diferentes tipos de imagens de entrada (as quais variam bastante no que diz respeito ao tipo de células que as compõem, densidade celular e distribuição dos diferentes níveis de intensidade da imagem), os resultados da aplicação de um algoritmo de detecção automática dos contornos celulares (realçados a verde) e as regiões delimitadas pelos contornos devidamente classificadas e representadas com uma cor distinta (neste caso, arbitrária) (retirado de [23]).

### 1.4.2.3 Filtragem Morfológica

As técnicas de filtragem morfológica podem ser definidas como transformações que ocorrem na imagem inicial, píxel-a-píxel, e que dependem não só do valor de intensidade do píxel sobre o qual ocorrem essas transformações, como também dos valores de intensidade dos seus píxeis vizinhos. O processo de filtragem é concretizado através da aplicação de matrizes, habitualmente também denominadas por máscaras ou *kernels*, à imagem inicial. Ao aplicar-se uma dada máscara centrada na posição  $(i,j)$  da imagem, sendo  $i$  o número de uma dada linha e  $j$  o número de uma dada coluna, o valor do píxel localizado nessa posição central é substituído por um novo valor que, tal como mencionado anteriormente, depende da intensidade dos píxeis inseridos na sua vizinhança e também dos pesos associados à máscara aplicada. Após terem sido transformados todos os píxeis da imagem de entrada, uma nova imagem é então gerada [35].

No processo de filtragem morfológica, contrariamente ao que se sucede no caso da extração de características, os filtros aplicados sobre a imagem são filtros não-lineares tais como o filtro morfológico de erosão, dilatação, abertura e fechamento. A aplicação deste conjunto de algoritmos, os quais podem facilmente ser combinados entre si ou sucessivamente aplicados de modo a gerar filtros potencialmente mais complexos e eficazes, permite analisar e manipular a morfologia dos objetos presentes na imagem.

### 1.4.2.4 Crescimento de Regiões por Acumulação de Píxeis

Este é um processo de segmentação bastante idêntico à limiarização, uma vez que se baseia na procura de valores de intensidade semelhantes em píxeis inseridos numa mesma vizinhança. A diferença mais significativa entre estes processos assenta no facto de que, no crescimento de regiões, a agregação de píxeis é inicializada em determinados pontos específicos da imagem, denominados por sementes. De um modo muito genérico podem definir-se três etapas principais que descrevem este método: a primeira consiste na escolha dos píxeis-sementes, a segunda na escolha do limiar de intensidade que separará as regiões e a terceira no crescimento dessas mesmas regiões [35].

A escolha dos píxeis-sementes representa uma etapa fulcral neste processo de segmentação uma vez que é a partir destas localizações que as regiões se irão desenvolver. O modo como a sedimentação é efetuada depende bastante do tipo de problema a ser abordado.

A escolha do limiar constitui também outro passo crucial. O valor escolhido para este parâmetro deve representar a diferença de valores de intensidade que desejamos que esteja compreendida dentro de cada uma das regiões.

A etapa do crescimento de regiões consiste no agrupamento iterativo dos píxeis pertencentes a uma dada vizinhança que se assemelhem em algum critério como por exemplo, a intensidade dos níveis de cinza [36].

Atualmente, um dos maiores problemas associado aos algoritmos de crescimento de regiões é que, normalmente, estes métodos produzem imagens sobre segmentadas, i.e. imagens repartidas em demasiadas regiões [35].

Um dos mais populares métodos de crescimento de regiões, inicialmente proposto por Digabel e Lantierjoul, é denominado por divisor de águas (ou em Inglês, *watershed*). Nos últimos anos, este método tem sido continuamente otimizado e adaptado a diferentes áreas de análise e segmentação de imagem. Segundo este algoritmo, os valores de intensidade são encarados como elevações numa paisagem hipotética e a segmentação é efetuada ao longo de regiões que se assemelham a zonas de drenagem dessa mesma paisagem, i.e. zonas da paisagem em que a elevação é mínima. Na correspondente imagem de gradiente, a “água” ergue-se a partir das áreas da imagem caracterizadas por uma menor magnitude até às zonas em que a magnitude de gradiente é máxima. Estas últimas correspondem às fronteiras das hipotéticas bacias hidrográficas ou *watersheds*. Uma variante deste método, bastante praticada, consiste em permitir com que o nível da “água” se erga a partir de determinadas zonas marcadas como sementes. O processo de sedimentação automática associada a esta técnica revela-se bastante complexo e conduz, frequentemente, a distribuições de sementes incorretas. Assim, de modo a contornar este problema, muitos dos algoritmos divisores de água optam por distribuir estas sementes manualmente – o que, apesar de conduzir a melhores resultados, transforma este processo numa abordagem de sedimentação excessivamente demorada e que requer demasiada interação por parte do utilizador [37].

#### 1.4.2.5 Curvas Deformáveis

A ideia básica intrínseca a esta técnica, como o próprio nome indica, consiste em definir-se uma curva paramétrica fechada, inserida no domínio da imagem, a qual, mediante a aplicação de um conjunto de forças externas e internas, sofre uma deformação de modo a ajustar-se às fronteiras de um dado objeto-alvo presente na imagem [38].

O objetivo da energia interna consiste em preservar a regularidade do contorno da curva, sendo por isso definida através de alguns parâmetros geométricos de contorno, tais como a área e a curvatura. Nestes termos é possível a incorporação de algumas informações conhecidas *a priori*, tal como a geometria das células em análise, o que contribui para redução e otimização dos resultados obtidos.

A energia externa, por sua vez, atua diretamente sobre o contorno inicial, atraindo-o para as regiões de fronteira do objeto. Esta componente energética é definida através de parâmetros associados a informações relativas à própria imagem, como por exemplo, a magnitude do gradiente da imagem de entrada. O processo de deformação ocorre sempre de modo a minimizar a energia total da curva inicial. O valor deste parâmetro é mínimo sempre que a curva se seja ajustada a uma região cujas características correspondam às pretendidas e previamente estabelecidas para a segmentação.

A definição adequada dos parâmetros que caracterizam a curva, a instabilidade do modelo sempre que a nitidez das bordas se revela insuficiente, bem como a automatização do processo de sedimentação representam apenas alguns dos maiores desafios, atualmente, associados a esta técnica de segmentação [23].

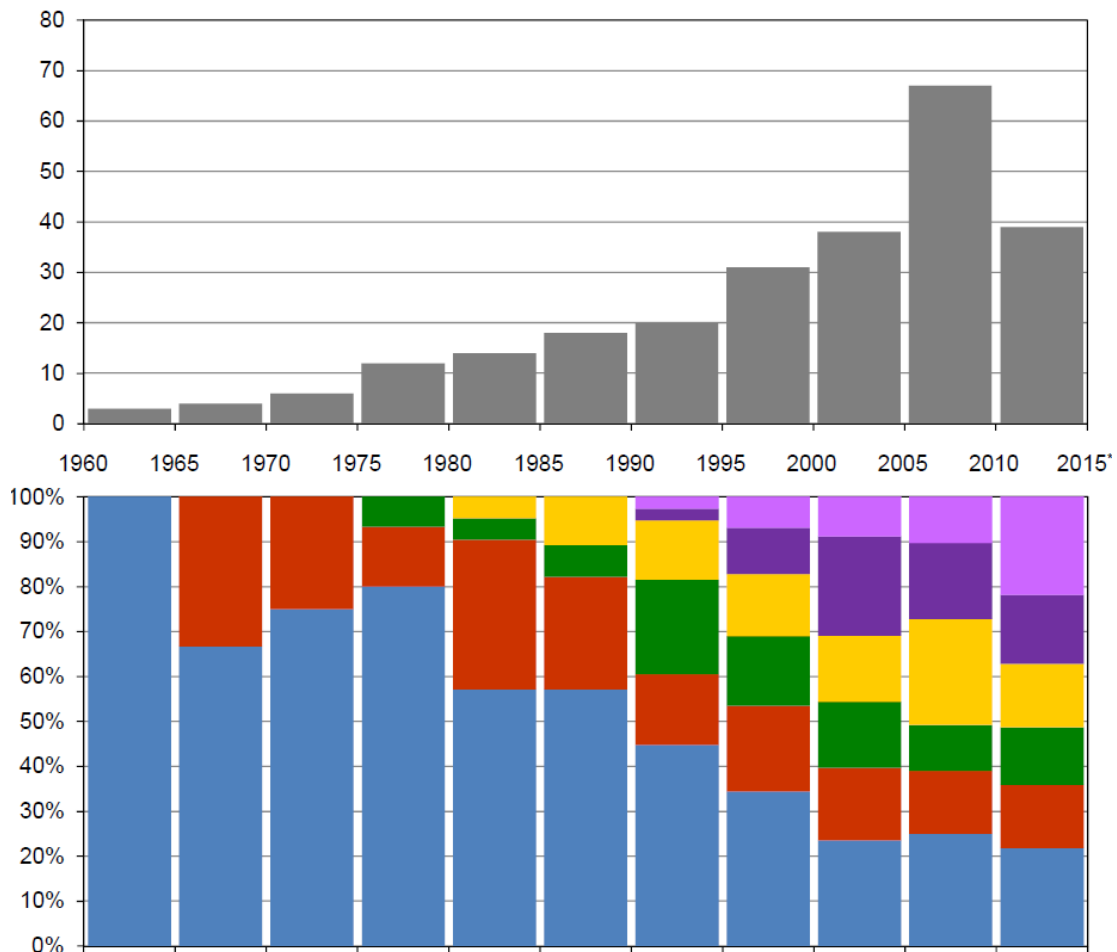
### 1.4.3 Análise estatística

Nos últimos 50 anos a quantidade de literatura e artigos desenvolvidos sobre a análise de imagens de células aumentou significativamente. A maioria das abordagens que foram publicadas foram primeiramente pensadas para servir outro tipo de propósitos e só posteriormente adaptadas à área da segmentação de células [23].

De modo a gerar o gráfico abaixo destacado, 250 artigos de jornal descrevendo diferentes métodos de segmentação e publicados em diferentes anos, foram analisados [23]. Com a informação providenciada por esta coleção de artigos, um histograma temporal, incluído na zona superior da **Figura 1.3**, foi desenvolvido. Este gráfico representa, para cada intervalo de tempo considerado, o número total de artigos publicados sobre esta temática. O asterisco associado ao último lustro ([2010;2015]) foi colocado para alertar para o fato do histograma ter sido elaborado apenas com informações recolhidas até Março de 2012 – daí o tamanho reduzido associado à última coluna do gráfico.

Na parte inferior da **Figura 1.3**, foi incluído um gráfico que representa a distribuição, em percentagem, dos diferentes métodos de segmentação incluídos nos artigos publicados em cada período de 5 anos representado. As principais categorias de algoritmos de segmentação abrangidos por este estudo são nomeadamente a limiarização (representada a azul), extração de atributos (representada a vermelho), filtragem morfológica (representada a verde), crescimento de regiões (representada a amarelo), curvas deformáveis (representada a roxo) e outras abordagens (repre-

sentada a violeta) – a qual inclui todos os métodos que não se inserem em qualquer outra categoria. Métodos de segmentação propostos ao longo das últimas décadas traduzem-se, geralmente, em combinações das abordagens acima mencionadas.



**Figura 1.3: Análise estatística da literatura publicada sobre segmentação de imagens de células.** No painel superior da figura é apresentado um histograma temporal do número de artigos publicados sobre a temática (o último lustro, destacado com um asterisco, foi constituído apenas com dados recolhidos até Março de 2012). No painel inferior, são destacadas as percentagens de artigos publicados, em cada período de 5 anos, sobre os diferentes tipos de abordagens de segmentação anteriormente mencionados. A azul encontra-se representando o método de limiarização de intensidade; a vermelho, a extração de características; a verde, a filtragem morfológica; a amarelo, crescimento de regiões; a roxo, curvas deformáveis e a violeta, outros tipos de abordagens (adaptado de [23]).

## 1.5 Objetivos da dissertação

A presente dissertação surge no âmbito de uma parceria de longa data entre o LBD e o *Computational Intelligence Research Group* (CA3) – um grupo de investigação criado em 1996 e que no ano de 2000 integrou o Centro de Tecnologia e Sistemas (CTS) da UNINOVA. O principal objetivo desta tese consiste em desenvolver e propor um novo algoritmo de segmentação a ser testado e aplicado a um conjunto de imagens da *Escherichia coli* obtidas por intermédio da microscopia de contraste de fase. A segmentação destas imagens, providenciadas pelo LBD, representa uma das etapas mais determinantes para o progresso de inúmeras investigações assumidas por este grupo de investigação. Atualmente, o método de segmentação que tem vindo a ser aplicado a esta coletânea de imagens exige um elevado grau de interação por parte do utilizador o que torna este processo bastante exaustivo e demorado. Deste modo, um dos mais importantes alvos associados a este novo algoritmo consiste em tornar este processo mais automático convertendo-o num procedimento não só menos demorado como também mais eficiente, simples e intuitivo do ponto de vista do utilizador.







## Fundamentação Teórica

### 2.1 Representação da Imagem Digital

Uma imagem monocromática pode ser encarada enquanto uma função bidimensional  $f(x, y)$  da intensidade luminosa, na qual  $x$  e  $y$  representam coordenadas espaciais, que por convenção variam entre:  $x = [1, 2, \dots, M]$  e  $y = [1, 2, \dots, N]$ . No ponto  $(x, y)$  o valor assumido por  $f$  representa a intensidade (ou brilho) da imagem nesse ponto, tal como ilustrado na **Figura 2.1**. Nesta figura foi efetuada a ampliação de uma dada região da imagem na qual se podem discernir os píxeis que a compõem, bem como as suas respectivas intensidades luminosas. A intensidade,  $I$ , de um determinado ponto de uma imagem é representada através de um valor inteiro, não-negativo e finito, frequentemente denominado por nível de cinzento. Sempre que  $x$ ,  $y$  e  $I$  assumirem valores discretos e finitos a imagem é denominada por digital [39].

A função  $f(x, y)$  é influenciada por dois parâmetros: a intensidade da iluminação da fonte que incide sobre objeto observado na imagem,  $i(x, y)$ , e a intensidade da iluminação refletida pelo mesmo,  $r(x, y)$ . Estas duas funções combinam-se num produto para originar  $f(x, y)$ :

$$f(x, y) = i(x, y) \times r(x, y)$$



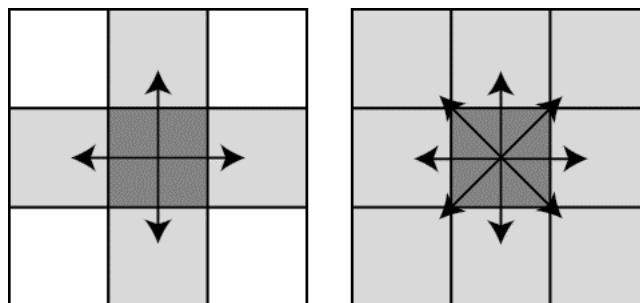
**Figura 2.1: Imagem digital monocromática “Goldhill” com destaque para uma região da imagem de 17×17 píxeis.** A imagem “Goldhill” é frequentemente utilizada para observar o resultado de testes e demonstrações de métodos de processamento de imagens (retirado de [39]).

## 2.2 O Píxel

O píxel (uma abreviatura para *picture element*) é o elemento estrutural da imagem digital. Frequentemente, este elemento é definido também enquanto o menor componente no qual uma imagem digital pode ser decomposta. Esta estrutura pode adotar diversas geometrias, no entanto, a mais comum corresponde à forma quadrangular. Este tipo de geometria conduz ao aparecimento de dois problemas que, por vezes, podem influenciar significativamente algumas das técnicas de processamento. O primeiro problema relaciona-se com o facto de o píxel ser anisotrópico, isto é, não apresentar as mesmas propriedades em todas as direções. Esta configuração faz com que este elemento possua dois tipos de vizinhança: uma vizinhança de borda e uma vizinhança de diagonal, ambas compostas por quatro píxeis, tal como demonstrado na **Figura 2.2**. Esta particularidade implica com que se tenha de definir o tipo de conectividade a ser aplicada: uma conectividade do tipo C4 (considerando apenas os píxeis presentes na vizinhança de borda ou de diagonal) ou C8 (incluindo os píxeis pertencentes a ambas as vizinhanças) [34].

O segundo problema surge como consequência direta do primeiro e relaciona-se com o fato de as distâncias entre um determinado ponto da imagem e a sua vizinhança depender da direção considerada. Este valor será igual a 1 para os vizinhos de borda e  $\sqrt{2}$  para os de diagonal [39].

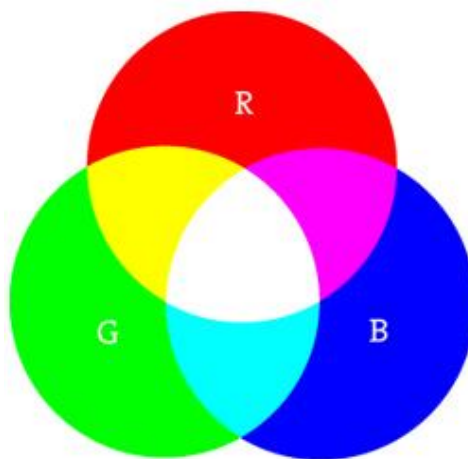
Operações morfológicas que utilizem máscaras (pequenas matrizes) do tipo 3x3, bem como operações de esqueletização são apenas alguns dos algoritmos sensíveis aos problemas anteriormente destacados. A solução para estes inconvenientes passa pela correção, mediante a aplicação de alguns ajustes ou ponderações, dos valores calculados através destas máscaras [34].



**Figura 2.2: Tipos de conectividade de píxeis.** A imagem localizada à esquerda na figura ilustra uma conectividade do tipo C4, através da qual o píxel central (representado com uma tonalidade mais escura de cinza) se encontra conectado vertical e horizontalmente aos quatro píxeis vizinhos de borda. A imagem localizada à direita na figura ilustra uma conectividade do tipo C8, através da qual o píxel central se encontra conectado a oito píxeis vizinhos, sendo quatro de borda e quatro de diagonais (adaptado de [40]).

Para representar imagens a preto e branco (ou mais corretamente dizendo, em tons de cinza) basta associar-se a cada píxel um valor numérico pertencente a uma dada escala de tonalidades. Caso optemos, por exemplo, por uma escala composta por 256 níveis de cinzento (variando entre o preto e o branco absoluto) é necessário associar-se a cada píxel da imagem um valor inteiro, não-negativo, de um *byte*. Convencionou-se, para este tipo de representação, que o valor 0 corresponderia ao preto e o 255 ao branco; os restantes valores numéricos compreendidos entre estes dois extremos representam níveis de intensidade intermédios na imagem.

Para representar imagens a cores, cada píxel é decomposto em três bandas: a primeira associada à cor vermelha, a segunda à cor verde e a última à cor azul, designadas em inglês por RGB (*red*, *green* e *blue*). Cada um destes componentes pode assumir um valor inteiro compreendido entre 0 e 255, indicando a intensidade de vermelho, verde e azul correspondentes a esse ponto da imagem. A combinação destes 3 componentes permite gerar um conjunto de aproximadamente 16,7 milhões de cores diferentes (ver **Figura 2.3**). Segundo este modelo, um píxel branco passará a ser representado por RGB = (255,255,255) enquanto um píxel preto passará a ser definido por RGB = (0,0,0) [41].



**Figura 2.3: Sistema de cores RGB.** Para se formar uma cor através deste modelo aditivo é necessário que os três componentes (vermelho, verde e azul) sejam sobrepostos. Adicionando o vermelho ao verde, gera-se amarelo; adicionando o azul ao vermelho, gera-se magenta; adicionando o azul ao verde, gera-se azul ciano; adicionando as três cores primárias, gera-se o branco (retirado de [42]).

## 2.3 Processamento de Imagem

O processamento de imagens digitais é, atualmente, uma área bastante explorada mas ainda promissora que inclui um conjunto de algoritmos desenvolvidos para a análise e manipulação de dados multidimensionais, de modo a realçar determinada característica ou extrair algum tipo de informação considerada útil na imagem.

Os sinais, como por exemplo as imagens, podem ser encarados como representações físicas que transportam determinada informação. Assim, processar uma imagem implica transformá-la progressivamente de modo a extrair corretamente a informação nela contida. Este processo, à semelhança do que ocorre com o sistema visual humano, dependendo do objetivo com que é aplicado, pode transformar-se num procedimento computacionalmente bastante complexo. Até à data não existe nenhuma solução única, suficientemente eficaz e abrangente para solucionar todos os problemas de processamento que têm sido apresentados [34]. Novas técnicas de processamento, mais complexas, têm sido desenvolvidas à medida que novos métodos, capazes de explorar automaticamente a informação contida nas imagens, foram sendo publicados. Este conjunto de métodos pode ser distinguido em duas categorias principais: a primeira inclui os métodos dedicados à análise da informação contida na imagem e a segunda abrange os métodos cujo objetivo consiste no enriquecimento da qualidade da imagem (em inglês, “*Image enhancement*”) [39]. Nos próximos parágrafos iremos discutir, de um modo bastante sucinto, cada uma destas vertentes.

### 2.3.1 Análise

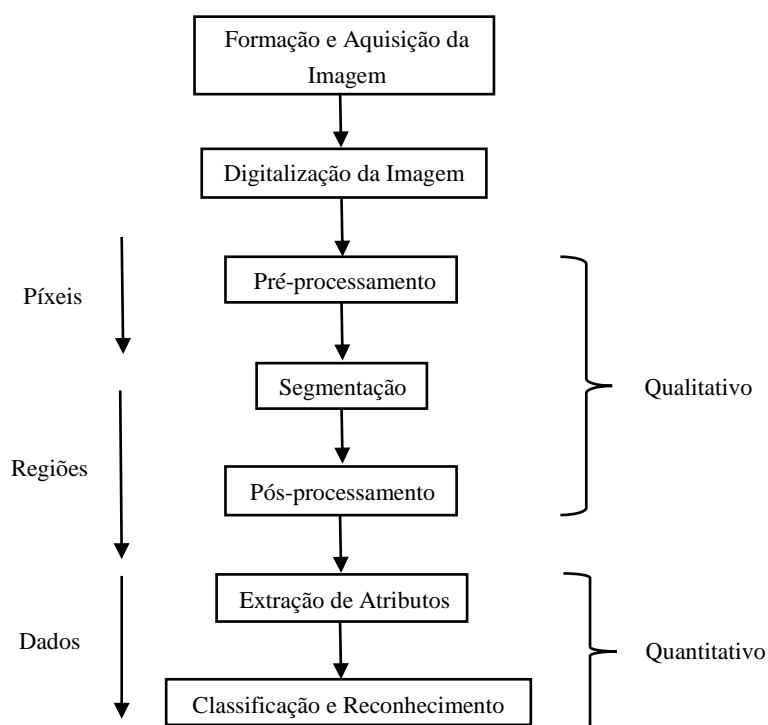
Este conjunto de técnicas de processamento visa, essencialmente, descrever a informação presente na imagem. Esta descrição baseia-se na extração, rápida e precisa, de uma série de medidas quantitativas (parâmetros) tais como a geometria das células, a área da sua superfície ou a densidade populacional presente numa determinada região. Algumas das aplicações mais comuns destes métodos de análise são, por exemplo, a contagem do número de células contidas numa dada amostra, a determinação do contorno das membranas celulares ou a descrição da distribuição espacial de uma determinada população de células. Estes métodos podem diferir bastante entre si, no que diz respeito à sua complexidade e à quantidade de tempo associada ao processamento.

### 2.3.2 Enriquecimento de Imagens

A expressão “enriquecimento”, no contexto do processamento de imagem, encontra-se associada à melhoria da qualidade da imagem inicial. Este tipo de transformação pode ser percebida pelo utilizador através do realce de determinada região da imagem ou, por exemplo, através de um aumento no contraste da mesma. A utilização de cores falsas, a correção das inomogeneidades na iluminação de fundo, bem como a aplicação de filtros espaciais são apenas alguns exemplos deste tipo de procedimentos. De um modo geral, estas técnicas atuam muito rapidamente e, sendo a maioria implementada em *hardware*, permitem ao utilizador fazer uma avaliação praticamente imediata das imagens processadas.

### 2.3.3 Etapas do Processamento de Imagem

O processamento de imagens pode ser visualizado enquanto um sistema constituído diversas etapas, entre as quais podemos salientar, de um modo sequencial, a formação e aquisição da imagem, digitalização, pré-processamento, segmentação, pós-processamento, extração de atributos, classificação e reconhecimento [39]. A ordem pela qual estas etapas são aplicadas, bem como outro tipo de informações relevantes inerentes a este processo encontram-se esquematicamente ilustradas na **Figura 2.4**. Posteriormente, irá apresentar-se uma descrição sucinta de cada uma destas etapas.



**Figura 2.4:** Etapas de um sistema de processamento de imagens (adaptado de [39]).

## 2.4 Aquisição e digitalização de imagens digitais

A aquisição de imagens é o processo através do qual um determinado cenário tridimensional (3D) é convertido numa imagem eletrónica (2D). Atualmente, o dispositivo de conversão mais utilizado é o CCD (do inglês, *charge coupled device*), o qual é composto por um circuito integrado contendo uma matriz de células fotossensíveis acopladas que, por funcionarem como condensadores, acumulam uma quantidade de carga elétrica proporcional à energia luminosa incidente.

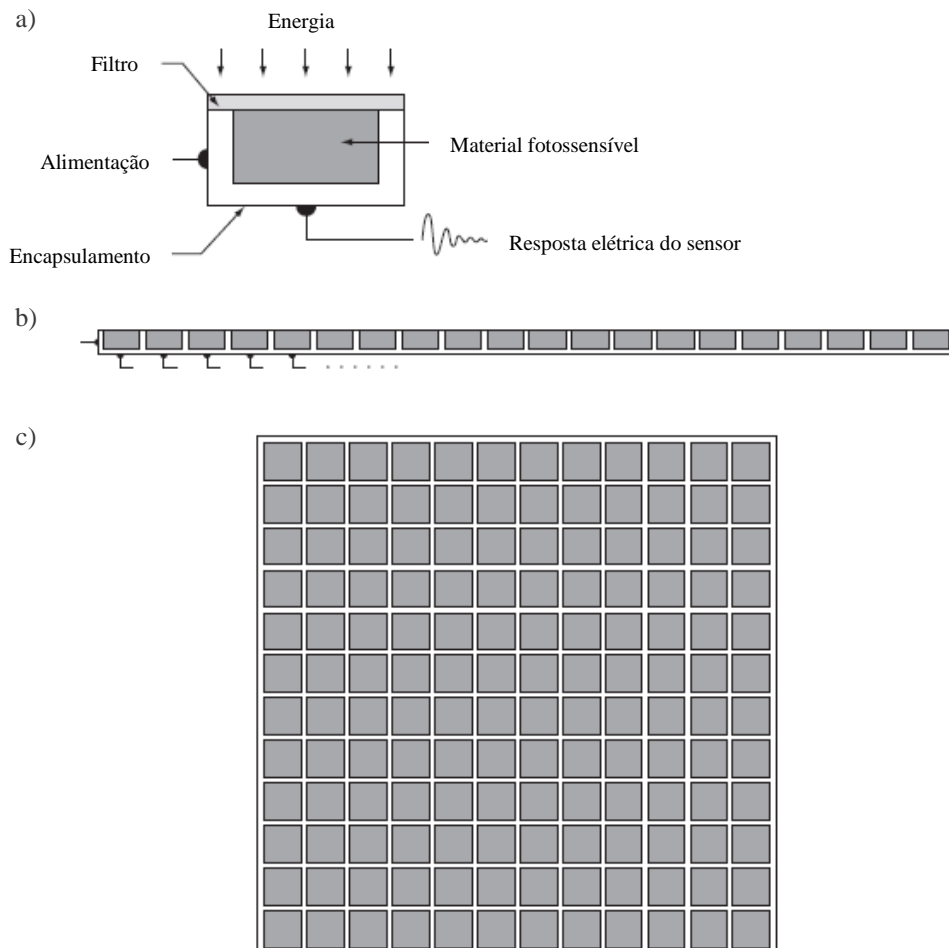
Para se efetuar a aquisição digital de imagens são necessários dois elementos cruciais. O primeiro, já exemplificado, é um dispositivo físico sensível a uma determinada gama de energias do espectro eletromagnético tal como a luz visível, ultra-violeta, infra-vermelha ou raios-x. Este dispositivo, para além de funcionar enquanto sensor, deverá desempenhar ainda a função de transdutor, produzindo à saída um sinal elétrico proporcional à intensidade da energia luminosa percebida [43].

Na parte superior (a) da **Figura 2.5** abaixo apresentada, encontram-se destacados alguns dos componentes mais importantes que constituem um sensor único de energia luminosa. Um dos sensores mais conhecidos deste tipo é o fotodíodo, um componente eletrónico semicondutor que

converte luz incidente em corrente elétrica. Tal como ilustrado na figura, este tipo de dispositivos possuem uma área de abertura ou conexão de fibra ótica, responsável por deixar passar a radiação incidente para a zona sensível do fotodíodo. A aplicação de um filtro à entrada da abertura permite seleccionar a gama de comprimentos de onda que pretendemos que atinja o material fotossensível.

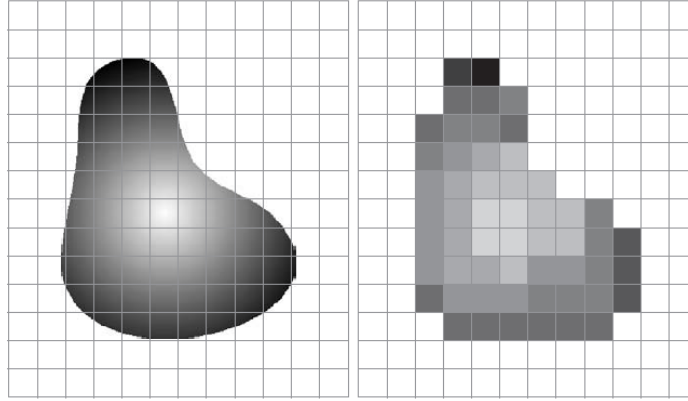
O arranjo linear de sensores, ilustrado em (b), traduz-se numa disposição geométrica aplicada com uma frequência bastante maior do que o modelo anteriormente apresentado. Este arranjo, por si só, apenas permite a aquisição de imagens ao longo de uma única direção (1D). Para adquirir uma imagem bidimensional (2D) é necessário que, durante a aquisição, este arranjo seja movido ao longo da direção perpendicular à linha formada pelos sensores.

Na imagem (c) encontra-se representado um conjunto de sensores individuais distribuídos ao longo de uma matriz bidimensional. Esse é o arranjo de sensores predominantemente encontrado nas câmaras digitais.



**Figura 2.5: Diferentes tipos de sensores de imagem.** (a) Sensor individual (fotodíodo). (b) Arranjo linear de sensores. (c) Matriz de sensores. (adaptado de [34]).

O segundo elemento-chave na aquisição de uma imagem consiste num digitalizador. Para que uma imagem,  $f(x, y)$ , possa ser armazenada e processada computacionalmente, é necessário que a mesma seja discretizada tanto ao nível das suas coordenadas espaciais como em intensidade (ver **Figura 2.6**). O processo através do qual ocorre a digitalização das coordenadas espaciais denomina-se por amostragem enquanto a digitalização dos valores de intensidade recebe o nome de quantização. A amostragem de  $f(x, y)$  ao longo das direções  $x$  e  $y$ , com  $y = 0, 1, \dots, N - 1$  e  $x = 0, 1, \dots, M - 1$ , resulta numa matriz de  $N \times M$  amostras. Cada elemento  $p(x, y)$  da imagem, i.e. cada píxel, poderá assumir o valor de um dos  $L$  níveis inteiros de cinza gerados por intermédio da quantização da imagem,  $\{0, 1, \dots, L - 1\}$ . Normalmente o limite inferior (0) do intervalo dos níveis de intensidade encontra-se associado à cor preta e o limite superior ( $L - 1$ ) à branca. Píxeis com valores compreendidos entre o 0 e o  $L - 1$  correspondem a diferentes tonalidades de cinza. A intensidade associada a estes píxeis será tão mais clara ou escura, quanto mais próximos estes valores estiverem do limite de intensidade superior ou inferior (respetivamente).



**Figura 2.6:** (a) Imagem contínua projetada numa matriz de sensores de imagem. (b) Resultado da aplicação dos processos de quantização e amostragem à imagem. (retirado de [34]).

A qualidade da imagem digital depende proporcionalmente dos valores associados a  $M$ ,  $N$  e  $L$ . Sabendo que os níveis de intensidade variam de acordo com a seguinte expressão:

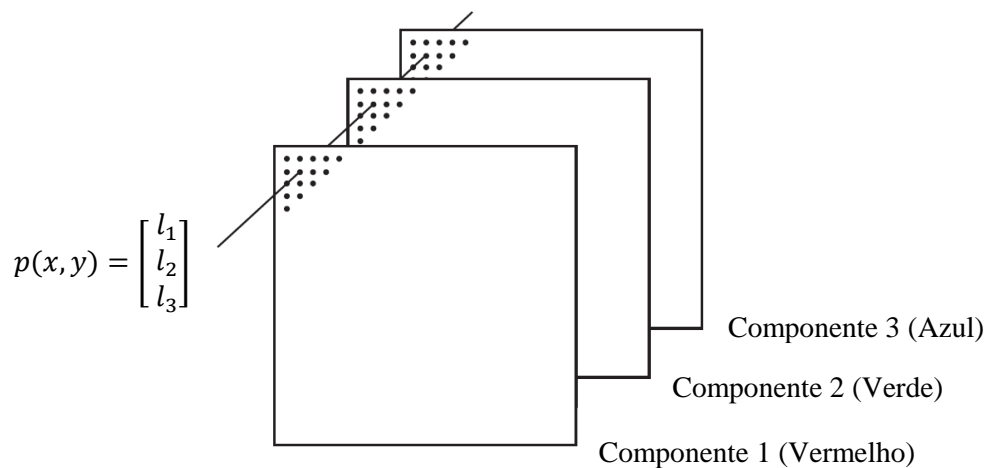
$$L = 2^l$$

com  $l \in \mathbf{N}$ , o número de bits,  $b$ , necessário para a representar e armazenar uma imagem digital do tipo  $N \times M$  é dado por:

$$b = M \times N \times l$$

No caso das imagens coloridas, cada píxel pode ser representado por um vetor do tipo  $p(x, y) = (l_1, l_2, l_3)$ , em que  $l_i$  é a intensidade de cada componente RGB e  $0 \leq l_i \leq L - 1$ . Esta representação encontra-se ilustrada na **Figura 2.7** abaixo destacada.





**Figura 2.7:** Formação do vetor  $p(x, y)$  a partir dos valores dos píxeis correspondentes, presentes nos três componentes RGB que compõem a imagem (adaptado de [34]).

## 2.5 Técnicas de Pré-Processamento

O principal objetivo do pré-processamento é melhorar a qualidade da imagem, corrigindo eventuais distorções proveniente da sua aquisição e/ou realçando determinadas características consideradas relevantes para as restantes etapas do processamento e análise [35]. Para que a etapa da segmentação produza resultados satisfatórios é fundamental que a imagem contenha o mínimo de imperfeições sendo, por isso, a etapa do pré-processamento considerada uma das mais importantes e influentes de todo o processo. As técnicas de pré-processamento podem ser distinguidas em duas categorias principais: métodos que operam no domínio da frequência e métodos que operam no domínio do espaço. Normalmente, para realçar determinados atributos, é aplicada à imagem uma combinação de métodos pertencentes a ambas as categorias. É durante o pré-processamento que maioria das transformações lineares e não-lineares são aplicadas à imagem. O melhoramento de contraste, correções na iluminação e a redução de ruído são apenas alguns dos exemplos mais comuns deste tipo de transformações [34].

### 2.5.1 Histograma

O histograma, também denominado por distribuição de frequências, é definido como uma representação gráfica de um conjunto de dados previamente tabulados e classificados. A distribuição destes elementos nas respetivas classes é representada, no histograma, através de colunas

ou retângulos cujas bases representam as diferentes classes e as alturas, as frequências (relativas ou absolutas) com que o valor associado a cada uma dessas classes ocorreu [39].

O histograma de uma imagem é, nada mais do que, uma representação gráfica do número de píxeis na imagem que apresentam um determinado nível de intensidade. Este gráfico é obtido através da contabilização do número de vezes que cada nível de intensidade aparece na imagem. Dividindo cada um destes resultados pelo número total de píxeis, obtêm-se uma distribuição da percentagem de cada nível de intensidade na imagem. A partir desta representação estatística é possível deduzir-se um conjunto de informações muito pertinentes relacionadas com a qualidade da imagem, nomeadamente informações relativas ao contraste, brilho e gama da mesma. Por outro lado, informações alusivas ao caráter espacial da imagem não se conseguem extrair a partir da análise desta representação gráfica. A probabilidade de um píxel na imagem apresentar um determinado nível de cinza é obtida, matematicamente, através da expressão abaixo destacada,

$$p_r(r_k) = \frac{n_k}{n}$$

em que:

$k = 0, 1, \dots, L - 1$ ;

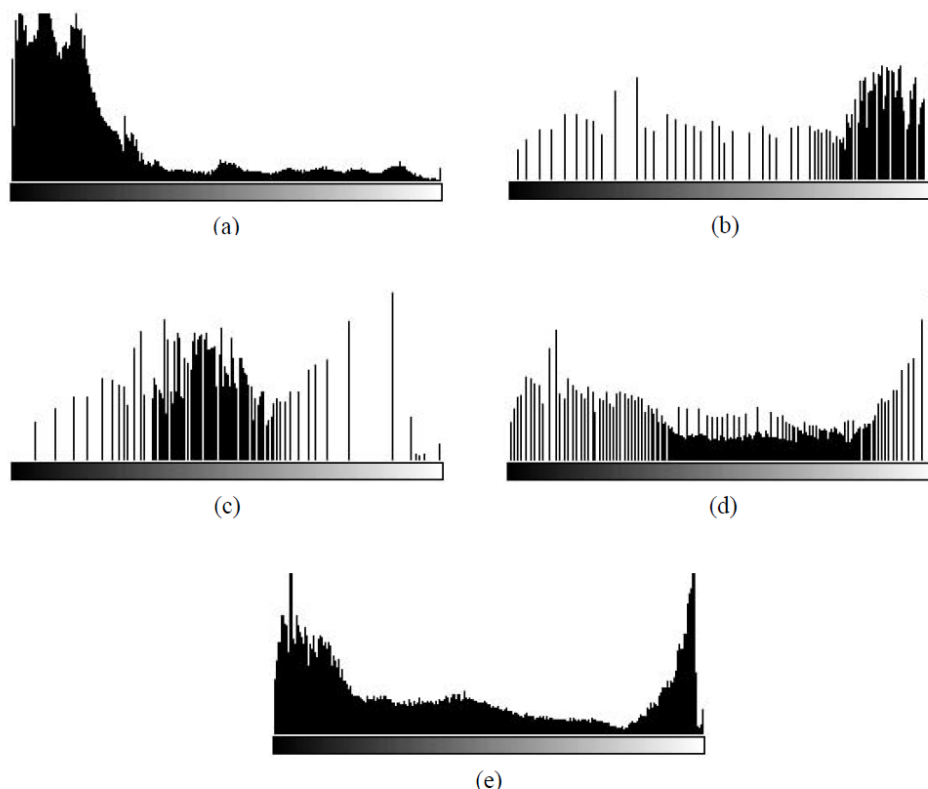
$L$  = número de níveis de intensidade da imagem digital;

$p_r(r_k)$  = probabilidade de ocorrência do  $k$ -ésimo nível de cinza;

$n$  = número total de píxeis na imagem;

$n_k$  = número de píxeis associados ao nível  $k$  de cinza.

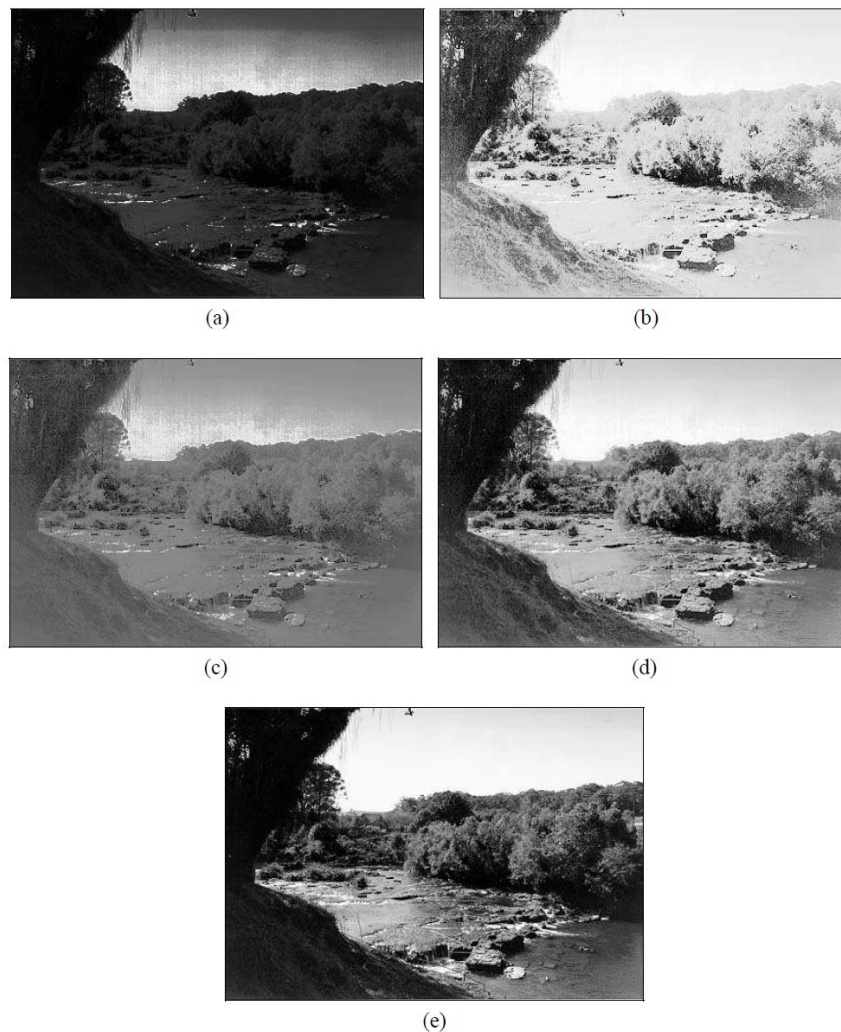
O histograma de uma imagem pode ser encarado enquanto uma função de distribuição de probabilidades a qual, por definição, consiste numa função que associa a cada valor previamente tabelado,  $L$ , da variável  $k$ , uma dada probabilidade de ocorrência,  $p_r(r_k)$ . Assim, considerando que o histograma obedece ao conjunto de propriedades e teoremas associados a este tipo de representação probabilística, pode concluir-se que  $\sum_k p(k) = 1$ .



**Figura 2.8:** Exemplos de histogramas (adaptado de [34]).

A **Figura 2.8**, acima destacada, apresenta cinco exemplos típicos de histogramas. As imagens monocromáticas correspondentes a estas distribuições de frequências encontram-se discriminadas na **Figura 2.9**, de modo a poder visualizar e analisar a relação que se estabelece entre estes dois conjuntos de representações. No primeiro histograma, distinguido na **Figura 2.8** (a), pode ser visualizada uma elevada concentração de píxeis associados a valores mais baixos na escala de intensidade. Este tipo de distribuição é característica de imagens predominantemente escuras. Na **Figura 2.8** (b), por sua vez, os píxeis encontram-se aglomerados na zona adjacente ao limite superior da escala de cinza – uma distribuição típica de imagens particularmente claras. Em (c), o agrupamento de píxeis ocorre a meio da escala, um reflexo de numa imagem caracterizada por um nível de brilho intermédio. Um elevado número de píxeis concentrado em zonas estreitas da escala de cinza, uma particularidade transversal às figuras **Figura 2.8** (a), (b) e (c), é próprio de imagens que possuem de um nível de contraste insuficiente. Esta característica surge, frequentemente, como consequência direta de alguns problemas técnicos relacionados com os dispositivos de aquisição de imagem ou como resultado de uma iluminação não-uniforme ou de baixa intensidade. No histograma representado na **Figura 2.8** (d) verifica-se que os píxeis da imagem correspondente se encontram, praticamente, distribuídos ao longo de toda a escala de

cinza. Este tipo de histograma, normalmente, corresponde a imagens que possuem um nível de contraste considerado próximo do ideal. Por último, na **Figura 2.8 (e)**, é apresentado, como exemplo, um histograma bimodal, isto é, um histograma no qual podem ser visualizados dois grupos de píxeis centralizados em dois pontos distintos no histograma: um deles em torno da região escura e outro em torno da região clara. Por se encontrarem consideravelmente espaçados, pode afirmar-se que a imagem correspondente apresenta um contraste elevado entre os dois grupos de píxeis.



**Figura 2.9:** Imagens correspondentes aos histogramas da Figura 2.8 (retirado de [34]).

O conceito de histograma pode ser aplicado também a imagens coloridas. Para tal é necessário que a imagem seja separada nos seus componentes RGB e que, para cada um deles, seja calculado o histograma correspondente. A análise do histograma permite extrair inúmeras informações qualitativas e quantitativas inerentes à imagem (tais como, a identificação do nível de

cinza mínimo, máximo e médio ou o reconhecimento de uma predominância de píxeis claros ou escuros). Outro tipo de informações qualitativas, tais como a detecção da presença ou ausência de ruído, só podem ser percebidas por intermédio da visualização e análise da imagem propriamente dita.

## 2.6 Operações pontuais, locais e globais

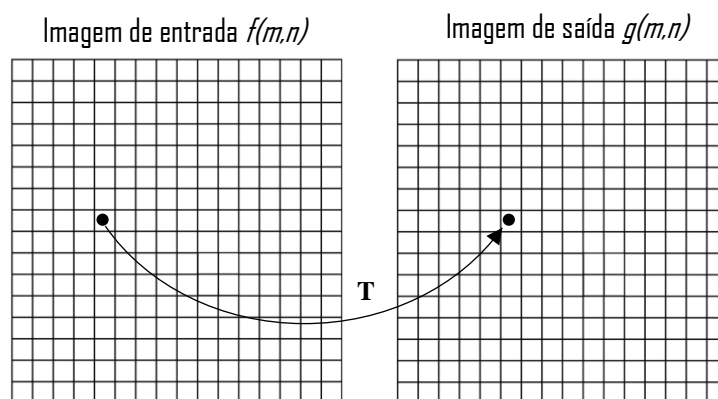
Os métodos de pré-processamento aplicados no domínio do espaço são executados através de operações matemáticas que atuam diretamente sobre os píxeis das imagens. Essas operações podem ser classificadas, consoante o tipo de vizinhança que consideram, em três categorias principais: operações pontuais, locais ou globais. Cada uma destas vertentes irá ser abordada sucintamente, em seguida.

### 2.6.1 Operação Pontual

Uma operação pontual (ver **Figura 2.10**) consiste numa transformação,  $T$ , aplicada à imagem mediante a qual, o valor de cada píxel é alterado individualmente, i.e., cada ponto da imagem de entrada,  $f(m,n)$ , gera, por intermédio de uma operação, um único ponto na imagem de saída,  $g(m,n)$ . Assim sendo, a dimensão da vizinhança considerada neste tipo de operações é de  $1 \times 1$ . As operações pontuais, por dependerem única e exclusivamente de valores pontuais de intensidade, são frequentemente representadas na forma:

$$s = T(r)$$

onde  $r$  representa a intensidade de  $f$  e  $s$  a intensidade de  $g$ , em qualquer ponto  $(m,n)$  correspondente a ambas imagens.



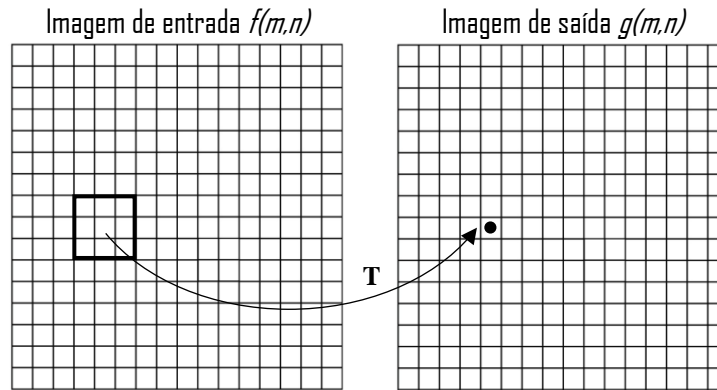
**Figura 2.10:** Representação da aplicação de uma operação pontual ( $T$ ) a um dado elemento  $(m,n)$  da imagem de entrada de modo a originar  $g(m,n)$ .

## 2.6.2 Operação Local

Uma operação local, ou filtragem, é uma transformação através da qual o valor de cada píxel é modificado de acordo com os valores de uma dada vizinhança pré-definida, tal como ilustrado na **Figura 2.11**. Dois exemplos deste tipo de transformações de imagem são a aplicação do filtro de média (filtro de suavização linear) ou do filtro de mediana (filtro de suavização não-linear). Ambas as transformações pertencem à categoria dos filtros passa-baixo e possuem como principal objetivo a suavização da imagem original (eliminação de ruído). Esta suavização será tanto maior quanto maior for a dimensão da máscara aplicada [44]. Uma máscara, ou *kernel*, consiste numa pequena matriz cujos elementos são habitualmente denominados por pesos ou coeficientes. Os valores assumidos por cada coeficiente definem o tipo de filtragem a que a imagem será submetida. Cada uma destas máscaras possui uma origem que, no caso das máscaras simétricas (i.e., máscaras com dimensões ímpares), é composta geralmente pelo píxel que ocupa a posição central. No caso das máscaras não simétricas (i.e., máscaras com dimensões pares), qualquer localização para o píxel de origem é válida encontrando-se esta escolha, normalmente, unicamente dependente do objetivo com que a máscara é aplicada.

O processo de filtragem, no domínio espacial, é concretizado através da convolução de uma máscara com a imagem. Depois de concluída esta operação, o valor de intensidade de cada píxel pertencente à imagem original passa a assumir um novo valor que, para além de depender do nível de intensidade inicial associado a este mesmo píxel, é também influenciado pela sua vizinhança. Algumas máscaras consideram como vizinhança apenas os píxeis abrangidos pelas direções vertical e horizontal; outras, por sua vez, para além destes, consideram também os píxeis incluídos ao longo das diagonais. No processo de convolução, cada coeficiente da máscara é multiplicado pelo valor correspondente na imagem  $f(x, y)$ . A soma desses resultados produz um novo valor de intensidade - uma média ponderada - que é atribuído à nova imagem,  $g(x, y)$ . Considerando um *kernel*,  $w$ , de dimensões  $n \times n$ , o processo de filtragem para cada píxel da nova imagem,  $g(m, n)$ , pode ser descrito por

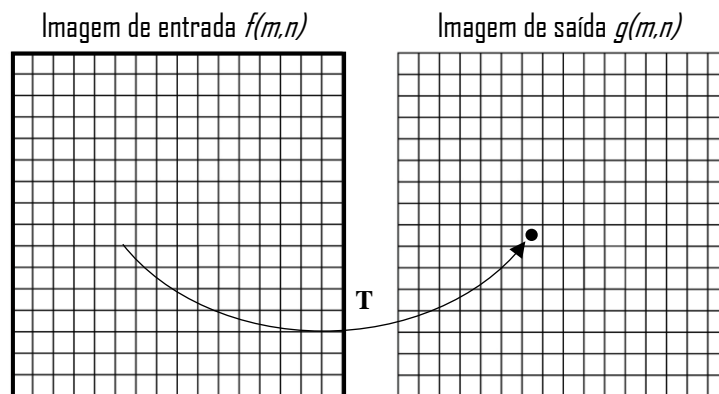
$$g(m, n) = \sum_{i=1}^{n^2} w_i \cdot f(x, y)$$



**Figura 2.11:** Representação da aplicação de uma operação local (T) à imagem  $f(m,n)$  de modo a originar  $g(m,n)$ . Neste exemplo, a dimensão da vizinhança considerada é de  $3 \times 3$  píxeis (representada a preto na figura).

### 2.6.3 Operação Global

Numa operação do tipo global, todos os píxeis da imagem de entrada são tidos em consideração para determinar o valor de um dado píxel da imagem de saída (ver **Figura 2.12**). Normalmente, métodos que atuam no domínio das frequências, como por exemplo a transformada de Fourier, incluem-se nesta categoria de operações. O histograma da imagem e a distância Euclidiana constituem outros dois exemplos, muito comuns, de operações do tipo global.

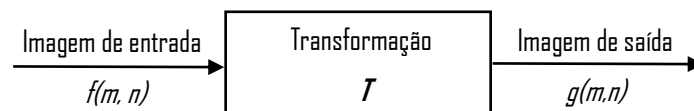


**Figura 2.12:** Representação da aplicação de uma operação global (T) à imagem  $f(m,n)$  de modo a originar  $g(m,n)$ .

## 2.7 Transformações de Intensidade

O pré-processamento de imagem requer, frequentemente, transformações na intensidade da mesma de modo a, por exemplo, tornar a imagem mais fácil de processar ou realçar determinado objeto de interesse. Transformações de intensidade (no domínio espacial) – tal como o próprio nome indica – são operações pontuais que atuam de forma a modificar os valores de intensidade dos píxeis da imagem, os quais se encontram, geralmente, compreendidos no intervalo:  $[0,255]$ . O mapeamento destes níveis de intensidade deve ocorrer de modo a gerar-se um segundo conjunto de valores também estes inseridos no intervalo:  $[0,255]$ .

As transformações de intensidade, expressas por  $g(m,n) = T[f(m,n)]$ , podem ser representadas visualmente pelo esquema da **Figura 2.13**, abaixo destacado.



**Figura 2.13:** Representação gráfica da aplicação de uma transformação de intensidade à imagem de entrada.

Estas transformações podem ser distinguidas, consoante o tipo de função que aplicam, em transformações lineares ou não lineares.

A transformação linear de uma imagem é uma operação pontual que mapeia o valor de intensidade de cada píxel noutra valor de intensidade através da aplicação de uma função linear. Por outro lado, uma transformação não-linear traduz-se numa operação pontual que executa esse mapeamento através da aplicação de uma função do tipo não-linear. Alguns exemplos, bastante comuns, de transformações pontuais lineares e não lineares irão ser, de seguida, apresentados.

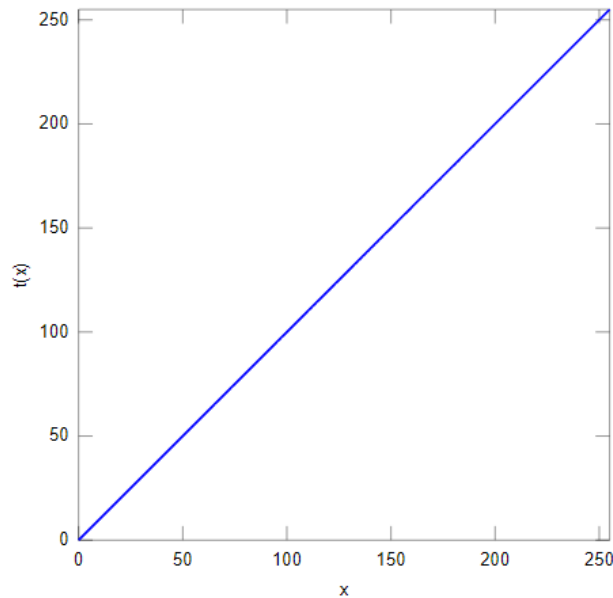
### 2.7.1 Função Identidade

A maneira mais simples de transformar uma imagem é mediante a aplicação da função identidade. Esta trata-se de uma função bijetiva, isto é, uma função que gera, para cada objeto do seu domínio, uma imagem igual ao próprio objeto. Esta função pode ser descrita matematicamente através da seguinte expressão:



$$f(x) = x$$

Graficamente, a função identidade é representada pela bissetriz dos quadrantes ímpares. Esta representação encontra-se ilustrada na **Figura 2.14**, abaixo destacada.



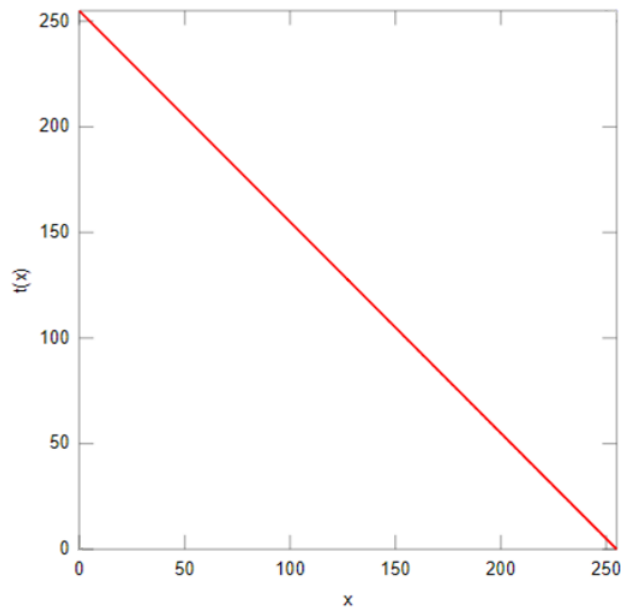
**Figura 2.14:** Representação gráfica do mapeamento resultante da aplicação da função identidade a cada elemento da imagem original. Neste tipo de gráficos,  $x$  representa o valor de intensidade de um dado píxel ( $m, n$ ) da imagem de entrada e  $t(x)$  o novo nível de intensidade gerado e atribuído ao píxel correspondente da imagem de saída (retirado de [45]).

### 2.7.2 Transformação Negativa

A transformação negativa de uma imagem consiste noutro exemplo, bastante simples, de uma transformação linear. Esta operação converte as zonas claras de uma imagem em zonas escuras, e vice-versa. Uma imagem negativa é obtida através da subtração do valor de intensidade de cada píxel a partir do valor correspondente ao limite máximo da escala de intensidades. Para uma imagem de 8-bits, esta transformação pode ser aplicada de acordo com a seguinte fórmula:

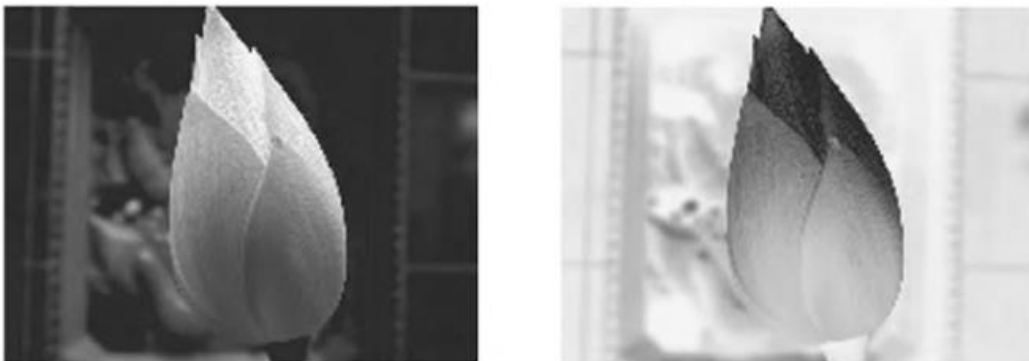
$$g(m, n) = 255 - f(m, n)$$

A representação gráfica deste mapeamento pode ser visualizada na **Figura 2.15**.



**Figura 2.15:** Representação gráfica do mapeamento resultante da aplicação da transformação negativa a cada elemento da imagem original (adaptado de [45]).

A partir da observação da **Figura 2.16**, é possível concluir que, por vezes, a imagem negativa pode providenciar mais informação do que a imagem original.



**Figura 2.16:** Resultado da aplicação da transformação negativa a uma imagem (retirado de [39]).

### 2.7.3 Ajuste de Contraste

O ajuste do contraste de uma imagem pode ser obtido através de inúmeros métodos computacionais. Na maioria das vezes esta transformação é executada por intermédio de uma opera-

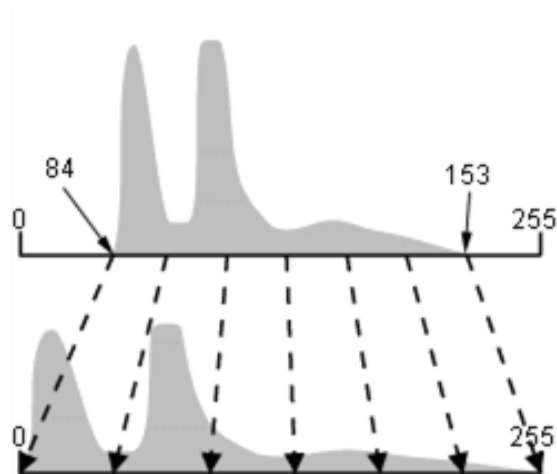
ção denominada por *stretching*, ou espalhamento de contraste. A forma mais simples de se implementar o *stretching* é através da aplicação de uma transformação linear cuja função de transferência consiste numa reta do tipo:

$$y = m \cdot x + b$$

O valor assumido por  $m$ , o declive da reta, é o fator que permite controlar o nível de contraste que queremos impor à imagem de entrada,  $f(m, n)$ . Na função de mapeamento apresentada,  $x$  corresponde ao valor de um dado píxel pertencente à imagem original e  $y$ , ao novo valor assumido por esse píxel após a transformação. A ordenada na origem,  $b$ , controla a intensidade global da imagem de saída. Sempre que o valor do parâmetro  $m$  for superior a 1, o contraste da imagem aumenta; sempre que for menor do que 1, o contraste da imagem diminui. Aumentar o contraste de uma imagem traduz-se, de um modo muito prático, em tornar as zonas escuras ainda mais escuras e as claras ainda mais claras. Esta alteração torna a análise e interpretação da imagem consideravelmente mais fácil.

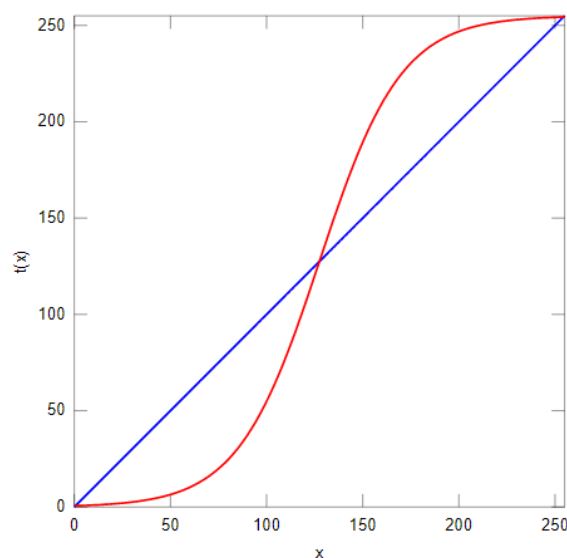
O objetivo subjacente deste método consiste na uniformização da distribuição das colunas que compõem o histograma da imagem original de modo a que estes elementos passem a preencher toda a gama do espetro de cinza. Esta operação implica, primeiramente, a identificação dos limites superior e inferior do histograma da imagem original para posteriormente expandi-los, de modo a abrangerem toda a escala de intensidades.

Na **Figura 2.17** abaixo destacada, encontra-se ilustrado um exemplo de *stretching* aplicado a um histograma com um limite inferior igual a 84 e um superior igual a 153. Os 70 níveis de intensidade, compreendidos entre o extremo máximo e o mínimo do histograma original, correspondem a uma ocupação inferior a um terço dos 256 níveis disponíveis. Através do espalhamento de contraste, esta gama de intensidades é expandida de modo a cobrir a toda a escala de cinza, a qual (para este exemplo, em específico) varia entre 0 e 255.



**Figura 2.17:** Exemplo da aplicação do método de *stretching* ao histograma de uma imagem. Antes de se ter efetuado o espalhamento do contraste, os valores de intensidade do histograma encontravam-se compreendidos entre 84 e 153. Depois do ajuste, estes valores passaram a ocupar toda a escala de cinza: [0; 255] (adaptado de [46]).

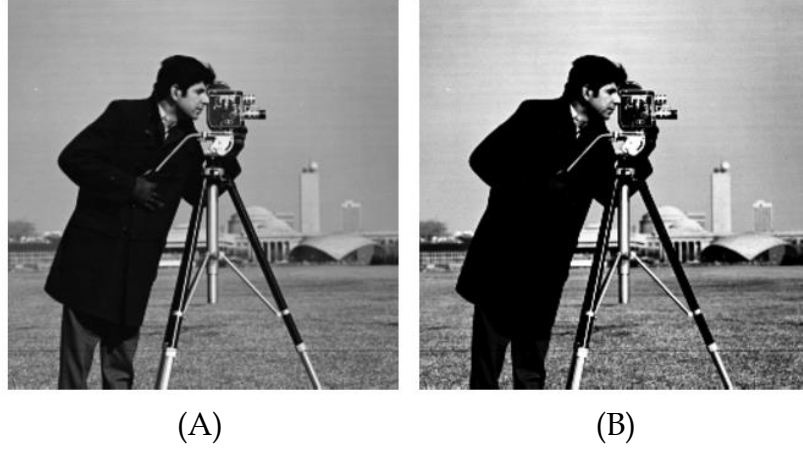
Outro método que pode ser utilizado para o ajustar o contraste de uma imagem consiste na aplicação de uma função sigmóide - representada a vermelho na **Figura 2.18** abaixo destacada. A denominação "sigmóide" surge devido à forma em S que caracteriza o seu gráfico.



**Figura 2.18:** Representação gráfica do mapeamento resultante da aplicação da função sigmoide a cada elemento da imagem original (retirado de [45]).

Ao mover esta função para a esquerda ou direita, a intensidade global da imagem resultante é alterada. O contraste pode ser manipulado através da variação da amplitude da função sigmóidal.

Na **Figura 2.19**, é introduzido um exemplo onde podem ser visualizados os efeitos da intensificação do contraste numa imagem através da aplicação desta função.



**Figura 2.19: Resultado da aplicação da função sigmóide no contraste de uma imagem.** (A) Imagem original; (B) Imagem após a aplicação de um ajuste sigmóidal no contraste (retirado de [45]).

A equalização do histograma é outro método de processamento que, à semelhança das duas técnicas anteriores, visa aumentar o contraste da imagem. Este ajuste é obtido através de uma redistribuição dos níveis de intensidade da imagem de modo a que estes passem a ocupar toda a escala de cinza. O objetivo é que esta distribuição seja equalizada, i.e., que todos os níveis de intensidade possuam, aproximadamente, o mesmo número de píxeis (originando um histograma idealmente plano). Por outras palavras, este método consiste numa operação não-linear que é aplicada a cada píxel da imagem de entrada,  $f(\mathbf{m}, \mathbf{n})$ , de modo a produzir uma distribuição uniforme dos níveis de intensidade na imagem de saída,  $g(\mathbf{m}, \mathbf{n})$ . O método mais frequentemente empregue para equalizar um histograma baseia-se na aplicação da função de distribuição acumulada (ou CDF, *Cumulative Distribution Function*). Esta operação pode ser descrita por:

$$s_k = T(r_k) = \sum_{j=0}^k \frac{n_j}{n} = \sum_{j=0}^k p_r(r_j)$$

onde:

$$0 \leq r_k \leq 1;$$

$$0 \leq s_k \leq 1;$$

$k = 0, 1, \dots, L - 1$  (em que  $L$  corresponde ao número total de níveis de intensidade definidos na etapa de quantificação da imagem);

$s_k$  representa os níveis de intensidade da imagem de saída;

$r_k$  representa os níveis de intensidade da imagem de entrada;

$n_j$  corresponde ao número de ocorrências do nível de intensidade  $j$  na imagem de entrada;

$n$  corresponde ao número total de píxeis da imagem de entrada;

$r_j$  representa o nível  $j$  da escala de intensidades da imagem de entrada;

$p_r$  corresponde à probabilidade de ocorrência de um dado nível de intensidade,  $r_j$ , na imagem de entrada;

Conhecendo a função de transformação aplicada (representada por  $T$ , na expressão acima discriminada), é possível reverter o processo e recuperar o histograma original. Para além desta vantagem, a equalização do histograma traduz-se, do ponto de vista computacional, numa operação muito pouco intensiva. Uma das principais desvantagens associadas a esta técnica assenta no facto de esta nem sempre ser discriminatória. Esta particularidade pode resultar, por vezes, numa ampliação do ruído de fundo e numa atenuação do sinal de interesse.

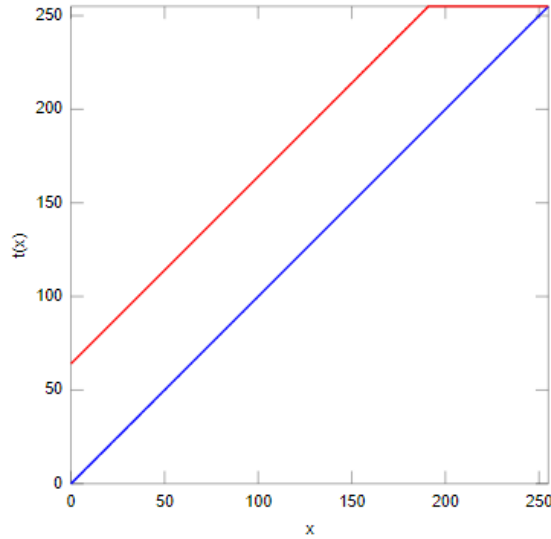
## 2.7.4 Ajuste de Brilho

O brilho global de uma imagem depende exclusivamente do valor de intensidade associado a cada píxel que a compõe. Este parâmetro pode ser modificado, simplesmente, através da adição ou subtração de uma constante,  $k$ , ao valor de cada amostra que constitui a imagem original,  $f(m, n)$ .

$$g(m, n) = f(m, n) + k$$

Sempre que a constante  $k$  for positiva, a imagem torna-se mais clara; caso contrário, a imagem escurece. O resultado desta transformação, tal como enfatizado na expressão X, é uma nova imagem,  $g(m, n)$ , sujeita à mesma escala de intensidades que a imagem original.

Caso o valor calculado para um determinado píxel ultrapasse o limite máximo (ou mínimo) da gama de tonalidades da imagem original, diz-se que ocorreu uma saturação de valores e, neste caso, o píxel passará a assumir o valor de intensidade do extremo que foi transposto. Este fenómeno encontra-se graficamente representado na **Figura 2.20**.



**Figura 2.20:** Representação gráfica do mapeamento que caracteriza o aumento do brilho (para  $k = 50$ ) de uma imagem (retirado de [45]).

### 2.7.5 Ajuste Gama

Tal como mencionado anteriormente, cada píxel de uma imagem possui um determinado nível de intensidade, também denominado por luminância. Este valor encontra-se compreendido entre 0 e 1, correspondendo o 0 à cor preta e o 1 à cor branca. Para que uma imagem digital possa ser observada é necessário um dispositivo de projeção. O que acontece, muitas das vezes, é que estes dispositivos (monitores, ecrãs de telemóvel ou televisões) não captam corretamente a luminância da imagem sendo, por isso, necessário aplicar-se uma correção a este processo [47].

A correção gamma, ou simplesmente gamma, é uma operação não-linear que permite controlar o brilho global de imagens projetadas. Este tipo de correção, nos casos mais simples, pode ser descrito através da seguinte lei de potências:

$$g(m, n) = A \cdot f(m, n)^\gamma$$

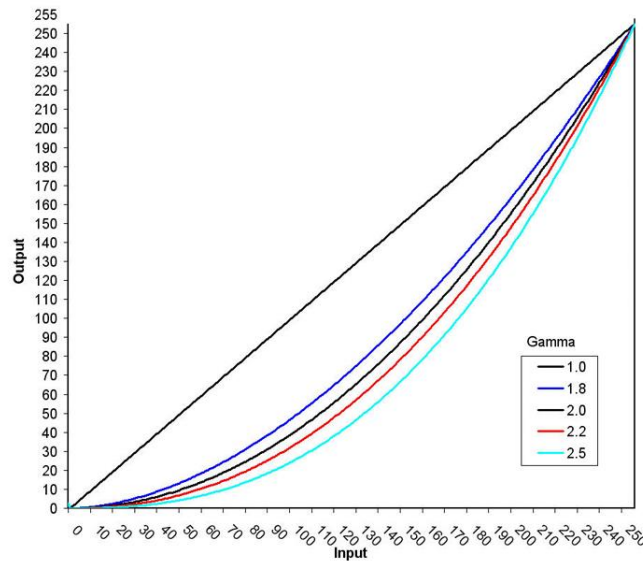
Através desta operação, cada valor da imagem de entrada,  $f(m, n)$ , é elevado à potência  $\gamma$  e multiplicado por uma constante,  $A$ , de modo a obter-se  $g(m, n)$ .

Imagens que não tenham sido devidamente corrigidas, no que diz respeito ao gamma, podem aparentar ser demasiado claras ou escuras. Uma modificação na quantidade da correção gamma aplicada, para além de influenciar o brilho global da imagem, altera também as proporções do vermelho, verde e azul.

A maioria dos monitores de computador, independentemente do fabricante, possui uma característica em comum: todos eles apresentam uma curva de intensidade (luminância ou brilho)

em função da tensão elétrica com um comportamento muito semelhante ao de uma função quadrática de ordem igual a 2,5. Na prática, isto significa que um píxel de intensidade  $x$  será projetado pelo monitor com uma intensidade igual a  $x^{2,5}$  [47].

Esta pequena divergência pode ser solucionada através da correção gamma. Conhecendo, a priori, a relação entre a tensão recebida pelo monitor e a intensidade por ele projetada, é possível corrigir-se o sinal antes que o mesmo seja recebido pelo monitor. Para tal, basta apenas elevar-se o sinal de entrada à potência inversa do gamma do monitor em questão (ver **Figura 2.21**).



**Figura 2.21:** Representação gráfica do mapeamento que caracteriza a correção gamma (para  $\gamma = [1.0; 1.8; 2.0; 2.2; 2.5]$ ) de uma imagem (retirado de [47]).

## 2.8 Segmentação

A segmentação, tal como mencionado anteriormente, é definida enquanto o processo através do qual uma imagem é repartida em zonas distintas caracterizadas por píxeis com atributos similares. Para ser significativa e útil para a análise e interpretação de imagens, estas regiões devem estar fortemente relacionadas com os objetos ou características de interesse. O sucesso da análise de imagem encontra-se inteiramente dependente desta etapa que, na maioria das vezes, constitui o primeiro e um dos mais importantes passos do processamento. Apesar da relevância e da evolução notória sentida ao longo do percurso dos algoritmos de segmentação, a obtenção de uma partição precisa continua, atualmente, a representar um dos problemas mais desafiadores inseridos no domínio da análise de imagem digital.



## Metodologia Adotada

### 3.1 MATLAB

O algoritmo de segmentação proposto foi desenvolvido com o auxílio da plataforma MATLAB. O MATLAB (uma abreviatura para **MAT**rix **LAB**oratory) trata-se de um *software* interativo, altamente direcionado para o cálculo numérico, que utiliza a matriz como elemento base da sua estrutura de dados. Esta aplicação informática, produzida pela empresa Norte-Americana *The Mathworks, Inc*, possui um conjunto vasto de funções já implementadas que se agrupam, consoante a área de aplicação, em bibliotecas denominadas por *toolboxes*. Contrariamente ao que se verifica na maioria das linguagens clássicas como Fortran, Basic ou C, no ambiente MATLAB não é necessário, entre outras tarefas habituais, a declaração de variáveis, utilização de ponteiros ou alocação de memória. Devido a estas, e muitas outras, facilidades providenciadas por este programa, a resolução de problemas relacionados com a análise numérica, cálculo com matrizes, processamento de sinais e construção de gráficos tornou-se bastante mais fácil, rápida e intuitiva [48].

Os comandos do MATLAB são, normalmente, diretamente digitados na Janela de Comandos (*Command Window*) e posteriormente executados pelo programa. No entanto, para além desta possibilidade, o MATLAB é também capaz de processar sequências de comandos contidas num ficheiro do tipo “.m”. Um algoritmo traduz-se em nada mais do que isto: uma sequência de comandos bem definida que visa atingir um determinado objetivo previamente estipulado [49]. Assim sendo, é necessário que, antes de se inicializar a etapa da implementação do algoritmo, se

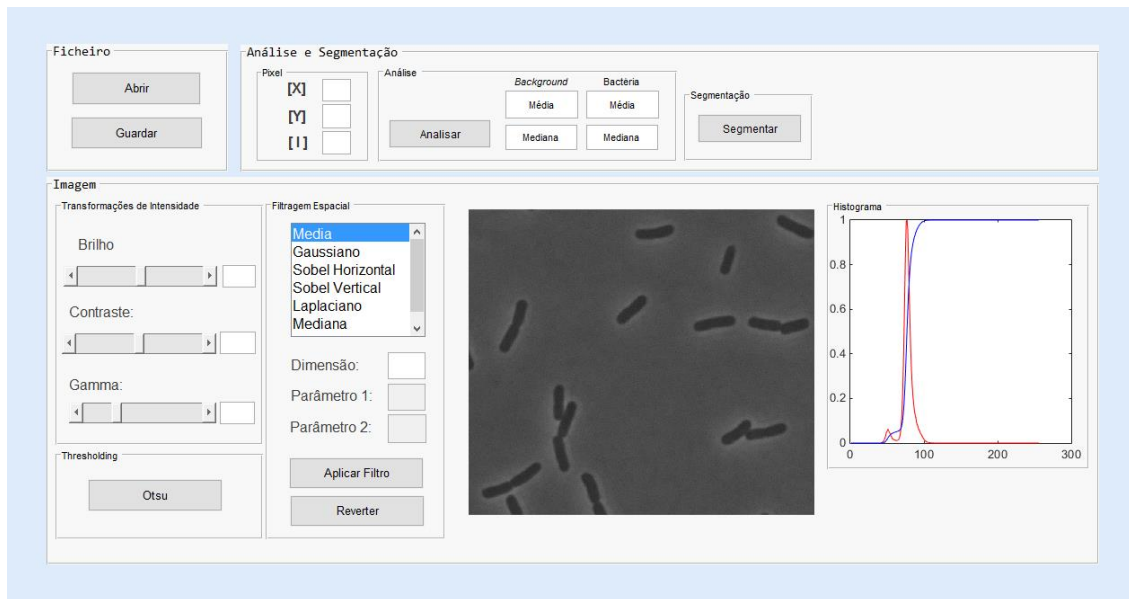
pondere, idealize e defina cada procedimento a adotar-se. Uma das ferramenta mais frequentemente aplicadas na esquematização de algoritmos é o fluxograma. Esta técnica, desenvolvida por Frank Gilberth em 1921, consiste numa representação gráfica bastante intuitiva das diferentes etapas que constituem um determinado processo em estudo [50]. Neste tipo de mapeamento, as atividades são normalmente representadas através de figuras geométricas (tais como círculos, retângulos, quadrados etc.), as quais possuem um significado específico associado. Estes símbolos, por sua vez, encontram-se interligados por setas, de acordo com o fluxo que caracteriza o processo [51].

Após a elaboração do fluxograma que descreve um determinado algoritmo, o próximo passo consiste em traduzir esse esquema para a linguagem de programação elegida, neste caso, o MATLAB.

Através do ambiente GUIDE (*graphical user interface development environment*) do MATLAB, é possível desenvolver-se, de um modo bastante rápido e intuitivo, a própria interface gráfica do utilizador (frequentemente abreviada por GUI, *Graphical User Interface*) [52]. Através desta ferramenta o utilizador pode comunicar com o programa de um modo bastante mais simples e intuitivo através da manipulação de inúmeros controlos gráficos tais como barras de ferramentas, menus, diferentes tipos de botões, *slidebars* e indicadores visuais. Estes elementos podem ser facilmente seleccionados e manipulados pelo utilizador através do rato e/ou do teclado. A construção do GUI divide-se essencialmente em duas etapas principais: uma delas, consiste em produzir o *design* do layout (o qual é arquivado num ficheiro do tipo “.fig”) e a outra, na programação das funções *callback* (guardadas num ficheiro do tipo “.m”, *M-file*). Estas funções são criadas automaticamente no *M-file* sempre que um novo componente é introduzido no GUIDE e devem discriminar o conjunto de operações que o utilizador pretende executar sempre que um determinado controlo for ativado [53].

## 3.2 Interface gráfica do utilizador

De modo a facilitar a compreensão da interface gráfica desenvolvida no decorrer desta tese (ver **Figura 3.2**) subdividiu-se a mesma, espacial e funcionalmente, em três zonas distintas: a primeira, a qual permite abrir e guardar o **Ficheiro** que se pretende segmentar; a segunda, através da qual é possível (entre outras funcionalidades) visualizar, filtrar e transformar a intensidade da **Imagem** seleccionada; a terceira, desenvolvida para a **Análise e Segmentação**.



**Figura 3.2:** Interface gráfica do utilizador desenvolvida no âmbito desta tese.

### 3.2.1 Abrir e guardar ficheiros

A primeira zona, localizada no canto superior esquerdo do GUI, permite ao utilizador seleccionar a imagem que pretende segmentar e guardar o resultado desse processo ou de outro tipo de operações às quais a imagem tenha sido sujeita. Estas duas funcionalidades foram implementadas, respetivamente, através da aplicação de dois conjuntos funções do MATLAB: `uigetfile` e `imread` para abrir a imagem; `uiputfile` e `imwrite` para guardar as alterações efetuadas.

Uma caixa de diálogo surge sempre que um dos botões pertencentes a esta região da interface é pressionado. No caso do botão “Abrir”, esta janela permite ao utilizador seleccionar ou introduzir o nome do ficheiro com o qual pretende trabalhar. Neste projeto em específico, como o tipo de ficheiros a processar são imagens, restringiu-se o conjunto de ficheiros considerados por este comando aos caracterizados pelas seguintes extensões: “\*.tif”, “\*.png” e “\*.jpg”. Cada um destes formatos de imagem possui um conjunto de propriedades muito específicas que os tornam mais indicados para diferentes circunstâncias. Depois de se seleccionar a imagem sobre a qual se irá trabalhar, a função `uigetfile` irá retornar o nome e caminho do ficheiro indicado. Caso, por algum motivo, o nome do ficheiro escolhido seja inválido, se prima sobre a opção “Cancelar” ou se encerre a caixa de diálogo, esta função retorna o valor 0. Partindo do pressuposto que este último cenário não se verifica, a imagem seleccionada irá ser lida pelo programa (com o auxílio da função `imread` do MATLAB) e projetada no sistema de eixos (axes1) localizado aproximadamente no centro da zona inferior da interface.

Depois de ter sido segmentada (ou simplesmente alterada) é possível, pressionando o botão “Guardar”, gravar-se o resultado da aplicação desses procedimentos no computador. Imediatamente após esse clique surge uma nova caixa de diálogo que permite ao utilizador definir o nome, diretoria e extensão com que pretende guardar a ficheiro. Depois destes parâmetros terem sido especificados, os seus valores são retornados pela função `uiputfile` e usados pelo comando `imwrite` para gravar a imagem com o nome, formato e diretoria previamente estipulados.

### 3.2.2 Visualização e processamento de imagens

Na segunda zona da interface, localizada na parte inferior do GUI, podem ser identificados quatro elementos principais: dois painéis e dois sistemas de eixos.

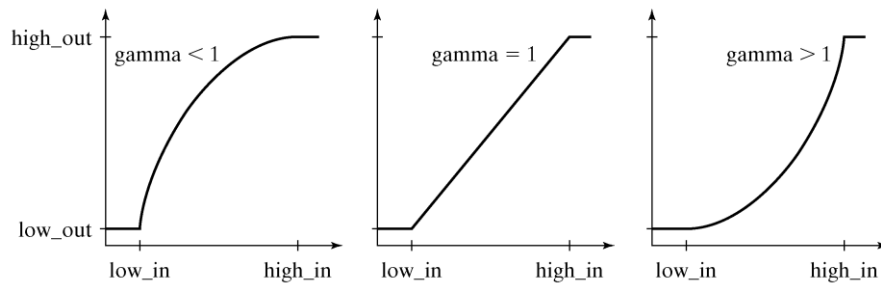
#### Transformações de Intensidade

O primeiro painel, o dos ajustes, permite (como o próprio nome sugere) ajustar o brilho, contraste e gamma da imagem, em qualquer fase do processamento. Cada um destes valores pode ser interactivamente ajustado pelo utilizador através da manipulação de uma *slidebar*. No *M-file*, estes ajustes são aplicados à imagem através da função `imadjust`. Esta função, pertencente à *toolbox* IPT (*Image Processing Toolbox*) do MATLAB, recebe quatro tipos de argumentos: o ficheiro sobre o qual irá ocorrer a transformação (`f`), o intervalo de intensidades da imagem de entrada (`[low_in high_in]`), o intervalo de intensidades da imagem de saída (`[low_out high_out]`) e o valor assumido pelo parâmetro gamma [54]. A sintaxe associada a função é dada por:

$$g = \text{imadjust}(f, [\text{low\_in } \text{high\_in}], [\text{low\_out } \text{high\_out}], \text{gamma})$$

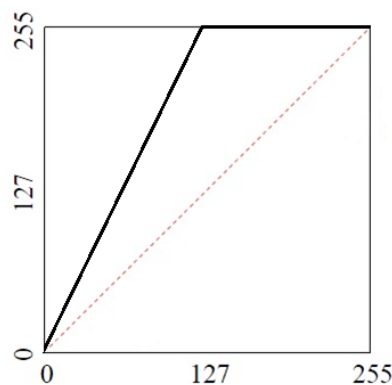
Através da aplicação deste comando é possível, tal como enfatizado na **Figura 3.2**, mapear os valores de intensidade da imagem original (`f`) em novos valores de intensidade a serem aplicados à imagem de saída (`g`). Nesta figura, verifica-se que o valor assumido pelo parâmetro gamma define curvatura da função de mapeamento. Nos casos em que gamma é menor do que 1, a intensidade da imagem aumenta (i.e., a imagem torna-se mais clara). Por outro lado, sempre que o gamma for maior do que 1, a intensidade da imagem diminui (i.e., a imagem torna-se mais escura). No caso específico em que gamma é igual à unidade, o mapeamento é linear e, caso não sejam aplicados outro tipo de ajustes, a imagem de saída será exatamente igual à imagem de entrada.

À exceção de  $f$  e  $\gamma$ , todos os elementos que compõem os argumentos da função `imadjust` (`low_in`, `high_in`, `low_out` e `high_out`) apresentam valores normalizados e por isso compreendidos entre 0 e 1.



**Figura 3.2:** Diferentes tipos de mapeamento associados ao ajuste do  $\gamma$  da imagem original através da aplicação da função `imadjust` (retirado de [55]).

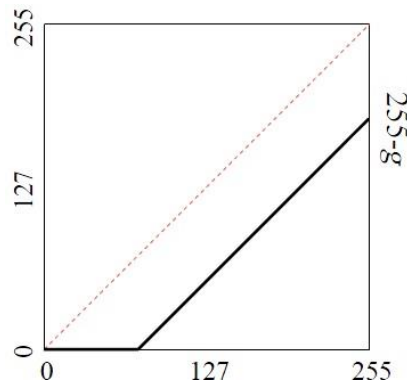
Para aumentar o contraste da imagem é necessário expandir-se o intervalo de valores de intensidade da imagem original. Um exemplo de uma função de transferência que visa o aumento do contraste é apresentado na **Figura 3.3**. Um parâmetro muito utilizado para ajustar este parâmetro é a inclinação da função de transferência aplicada. Sempre que a inclinação, relativamente ao eixo horizontal, for superior a  $45^\circ$ , o contraste da imagem aumenta. Por outro lado, sempre que a declive desta função for inferior a  $45^\circ$ , o contraste da imagem diminui.



**Figura 3.3:** Mapeamento correspondente a um aumento do contraste da imagem original (adaptado de [56]).

Variações no brilho global de uma imagem traduzem-se em translações horizontais na função de transferência. Considerando, como referência, a função característica do mapeamento linear (representada a rosa tracejado na **Figura 3.4**), podemos verificar que a reta representada a

preto nessa mesma figura consiste numa translação horizontal da primeira. Sempre que a translação horizontal ocorrer da esquerda para a direita, o brilho da imagem diminui. Contrariamente, sempre que a translação ocorrer da direita para a esquerda, o brilho da imagem aumenta.



**Figura 3.4:** Mapeamento correspondente a uma diminuição do brilho da imagem original (retirado de [56]).

Através da compreensão do comportamento das funções de transferência aquando da execução de um determinado tipo de ajuste (contraste, brilho ou gamma), foi possível determinar-se o conjunto de condições a atribuir a cada argumento da função `imadjust` de modo a que estas transformações pudessem ocorrer de um modo interativo.

Tal como mencionado anteriormente, ao manipular-se cada uma das *slidebars* incluídas nesta área da interface, é possível ajustar o valor do brilho, contraste e gamma da imagem original. Cada um destes controlos é caracterizado pelo intervalo de valores que o parâmetro que descreve pode assumir e pelo valor com o qual deverá ser inicializado. No caso do brilho e contraste, estas estruturas são inicializadas com o valor 0 e apresentam uma gama de valores compreendida no intervalo:  $[-1;1]$ . No caso do parâmetro gamma, a *sliderbar* é inicializada a 1 e apresenta uma gama de valores que varia entre 0.3 e 3.

Ao alterar-se a posição de cada *sliderbar* é possível visualizar-se o valor correspondente a essa mesma posição no indicador visual (*edit text*) colocado à direita de cada barra ajustável. Imediatamente depois de ser projetado, este valor é lido e utilizado como parâmetro de entrada da função `imadjust`. Nesta etapa, foi necessário estabelecer-se inúmeras condições de modo a poder abranger-se os diversos cenários possíveis. Para além de se ter de definir, na própria sintaxe da `imadjust`, as condições que conduzem a um aumento ou diminuição do brilho, contraste e gamma foi necessário especificar-se ainda, fora desse comando, como proceder nos casos em que um destes parâmetros é incrementado e posteriormente reduzido (ou vice-versa). Inicialmente,

esta ferramenta tinha sido desenvolvida de tal modo que os ajustes, obrigatoriamente, teriam de ser aplicados sequencialmente de acordo com uma ordem previamente estabelecida (primeiro teria de se efetuar o ajuste do brilho, depois do contraste e por último do gamma). Para além deste inconveniente, este algoritmo falhava sempre que a barra era deslocada num sentido e posteriormente movida no sentido oposto. Depois de algumas alterações significativas no código, foi possível otimizar-se esta aplicação de modo a:

- Poder ajustar-se a imagem de acordo com a ordem (e número de vezes) que parecer mais conveniente ao utilizador;

- Poder inverter corretamente o sentido da aplicação dos ajustes.

## Filtragem espacial

O segundo painel, o da filtragem, permite transformar uma dada imagem através da aplicação única ou sucessiva de filtros espaciais. Neste conjunto de operadores foram incluídos, nomeadamente, o filtro de média, mediana, gaussiano, Laplaciano e Sobel (vertical e horizontal).

Um dos principais problemas que afetam a qualidade de uma imagem digital consiste na contaminação da mesma com ruído. O ruído é definido, de um modo genérico, como um conjunto de informações indesejáveis que se sobrepõem à informação contida nas imagens. Os dois tipos de ruído mais frequentemente detetados nas imagens são nomeadamente o *Salt and Pepper* (Sal e Pimenta) e o ruído Gaussiano. Este último é um ruído estatístico que apresenta uma distribuição de Gauss e que é normalmente originado por uma má iluminação, altas temperaturas ou problemas associados à transmissão da imagem. O ruído *Salt and Pepper*, normalmente, surge como consequência de uma má transmissão de dados. Para atenuar tais ruídos, ao longo dos anos, foram desenvolvidos inúmeros métodos de suavização. Um dos principais objetivos da aplicação destas técnicas consiste em preparar as imagens para etapas posteriores do processamento, como por exemplo a segmentação. A aplicação de filtros capazes de atenuar o ruído presente numa imagem e simultaneamente preservar as suas características é fundamental e deve ser incluída na etapa do pré-processamento. De um modo geral, os métodos desenvolvidos para a atenuação do ruído Gaussiano são menos eficientes que os métodos desenvolvidos para a atenuação do ruído impulsivo (ou sal e pimenta). Dentre as técnicas mais conhecidas de suavização destacam-se a aplicação do filtro de média, mediana e gaussiano. Estes três métodos serão posteriormente descritos bem como a aplicação do filtro Laplaciano e Sobel (filtros de realce).

### Filtro de Média

Os filtros de média são filtros simples aplicados frequentemente na etapa de pré-processamento de imagem para a redução de ruídos.

Através desta técnica de filtragem, o valor de cada píxel da imagem de entrada é substituído pela média aritmética dos valores de intensidade dos píxeis pertencentes a uma dada vizinhança (ver **Figura 3.5**). Quanto maior for o tamanho da máscara considerada, maior será a perda dos pequenos detalhes da imagem resultante, efeito conhecido por *blurring* ou desfocagem. De modo a atenuar este efeito, diferentes tipos de ajustes a este algoritmo têm sido propostos. Uma dessas otimizações consiste em comparar o resultado obtido com um determinado limiar ( $T$ ) antes de se proceder à alteração da intensidade. O filtro de média pode ser desenvolvido através da função `fspecial` do MATLAB e aplicado à imagem através da função `imfilter`.

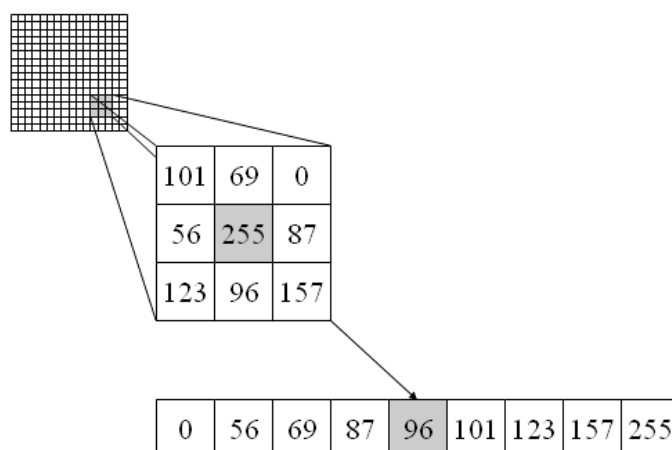
$$\frac{1}{9} \times \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

**Figura 3.5:** Filtro de média representado por uma máscara do tipo 3×3 (adaptado de [57]).

### Filtro de Mediana

O filtro de mediana é um dos filtros de suavização mais frequentemente considerados para a eliminação do ruído impulsivo em imagens digitais. Este filtro não-linear atua substituindo o valor de intensidade de cada píxel da imagem original pela mediana dos valores de intensidade dos píxeis inseridos numa dada vizinhança (ver **Figura 3.6**). O valor mediano é o valor que ocupa a posição central de um dado conjunto de elementos organizado por ordem crescente (ou decrescente) de valores. Sempre que o número de elementos que compõem esse conjunto é par, a mediana é dada pela média dos dois elementos que ocupam a zona central. Uma vantagem deste método, comparativamente ao filtro de média, é que através da aplicação deste tipo de filtragem os detalhes da imagem original são melhor preservados. Computacionalmente, este processo de filtragem é aplicado à imagem através da função `medfilt2`.





**Figura 3.6:** Representação gráfica de como computar a mediana de uma dada região da imagem (adaptado de [58]).

### Filtro Gaussiano

A filtragem Gaussiana é uma técnica de suavização de imagem frequentemente aplicada na etapa do pré-processamento. A curva de Gauss, também conhecida como distribuição Normal ou função de Gaussiana, é uma das funções mais importantes da matemática e da estatística. Esta distribuição, quando considerada a duas dimensões, pode ser utilizada para definir os pesos da máscara de convolução associada tipo de filtragem. A função de Gauss, a duas dimensões, é dada pela seguinte expressão:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

O resultado da convolução desta máscara com a imagem original será tão mais eficiente quanto maior for a largura da curva de Gauss utilizada na construção ao *kernel*. A largura desta curva é definida pelo valor do desvio padrão ( $\sigma$ ) aplicado.

Este filtro, à semelhança de outros, atua substituindo o cada píxel da imagem original pela média ponderada dos píxeis incluídos numa dada vizinhança. A grande diferença, relativamente a outros métodos, é que os peso atribuído a um determinado píxel da vizinhança decresce à medida que a distância entre este e o píxel central aumenta.

Comparativamente ao filtro de média, verifica-se que este tipo de filtro, por possuir pesos com valores distintos, preserva melhor os contornos da imagem original. Na **Figura 3.7** é apresentado um exemplo de uma máscara típica de um filtro Gaussiano (neste caso específico, com

$\sigma$  igual a 1.4). O filtro gaussiano, à semelhança do filtro de média, Laplaciano e Sobel, é obtido computacionalmente através da função `fspecial` do MATLAB e aplicado à imagem através da função `imfilter`.

$$\frac{1}{159}$$

2	4	5	4	2
4	9	12	9	4
5	12	15	12	5
4	9	12	9	4
2	4	5	4	2

**Figura 3.7:** Exemplo de uma máscara de convolução (5×5 píxeis) para a aplicação do filtro Gaussiano (com  $\sigma = 1.4$ ) (retirada [59]).

### Filtro Laplaciano

O Laplaciano  $L(x,y)$  de uma função bidimensional  $f(x,y)$  consiste na sua derivada de segunda ordem, a qual é definida por:

$$L(x,y) = \frac{\delta^2 f}{\delta x^2} + \frac{\delta^2 f}{\delta y^2}$$

Esta equação pode ser convertida numa máscara, por exemplo do tipo 3×3, igual ou semelhante às matrizes apresentadas na **Figura 3.8**.

0	1	0
1	-4	1
0	1	0

1	1	1
1	-8	1
1	1	1

-1	2	-1
2	-4	2
-1	2	-1

**Figura 3.8:** Três exemplos comuns de máscaras de convolução (3×3) para a aplicação do filtro Laplaciano (retirada de [60]).

Sendo o Laplaciano uma derivada, a soma dos seus coeficientes terá de ser obrigatoriamente nula. Normalmente o coeficiente associado ao píxel central da máscara é negativo e os

restantes píxeis positivos no entanto, tal como podemos observar através dos exemplos acima destacados, a situação inversa e outras variantes também são igualmente válidas.

No caso em que  $f(x,y)$  é uma imagem, a convolução com este filtro irá substituir o valor de cada píxel da imagem original pela média ponderada dos píxeis vizinhos. A aplicação do filtro Laplaciano irá atuar de modo a realçar as regiões da imagem onde ocorrem mudanças bruscas de intensidade, sendo por isso o Laplaciano normalmente utilizado para a deteção de contornos.

## Filtro Sobel

O filtro Sobel é um operador frequentemente aplicado no processamento de imagem que, à semelhança do filtro Laplaciano, visa realçar os contornos da imagem original

Este filtro calcula, para cada píxel, o gradiente de intensidade da imagem nesse ponto. Este parâmetro consiste num vetor que é definido por uma direção (da maior variação de claro para escuro) e pela quantidade de variação que ocorre ao longo dessa direção. Como os limites dos objetos presentes na imagem, geralmente, correspondem a zonas de variações intensas claro-escuro, este algoritmo consegue facilmente detetar os contornos da imagem.

A aplicação deste filtro à imagem original ocorre através da convolução desta imagem com duas matrizes do tipo  $3 \times 3$ , as quais se encontram destacadas na **Figura 3.9**.

$$G_x = \begin{array}{|c|c|c|} \hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline \end{array} \quad G_y = \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 0 & 0 & 0 \\ \hline -1 & -2 & -1 \\ \hline \end{array}$$

**Figura 3.9:** Máscaras ( $3 \times 3$ ) utilizadas para a aplicação (vertical e horizontal) do filtro Sobel (adaptado de [61]).

Enquanto a matriz  $G_x$  é utilizada para detetar as variações horizontais na intensidade da imagem, a matriz  $G_y$  deteta as variações verticais.

Por sua vez, a magnitude,  $G$ , e a direção,  $\theta$ , do gradiente são definidos por:

$$G = \sqrt{G_x^2 + G_y^2}$$

$$\theta = \arctan\left(\frac{G_y}{G_x}\right)$$

## Sistemas de eixos

O sistema de eixos apresentado imediatamente à esquerda do painel de filtragem, tal como mencionado anteriormente, permite visualizar a imagem inicialmente selecionada e o resultado das suas respetivas alterações. Por sua vez, o sistema de eixos localizado à direita do primeiro foi programado de modo a projetar, a cada instante, o histograma da imagem em processamento.

### 3.2.3 Análise e Segmentação

A terceira zona da interface foi desenvolvida para a aplicação da segmentação às imagens. Neste painel foram incluídos alguns elementos que, embora não se relacionem diretamente com o algoritmo desenvolvido, foram essenciais, ao longo do processo, para analisar determinadas características da imagem.

O primeiro conjunto de elementos a ser considerado, apresentado no painel do “Píxel”, permite ao utilizador visualizar, através de um simples clique, os valores das coordenadas espaciais e o nível de intensidade associados a um determinado píxel de interesse.

O segundo conjunto de elementos, destacado no painel da “Análise”, permite calcular e visualizar o resultado da média e da mediana dos valores de intensidade associados a uma determinada área da bactéria (de dimensões pré-definidas e posição ajustável) e de uma área retangular do *background* (de posição e dimensões ajustáveis). A partir destes parâmetros é possível analisar-se, de um modo mais prático e rápido, a forma como os diferentes níveis de intensidade se distribuem na imagem.

Inicialmente, esta ferramenta foi bastante útil para perceber, aproximadamente, quais os níveis de intensidade que constituíam o plano de fundo da imagem, as bactérias ou qualquer outra região considerada relevante para o estudo.

Para executar este conjunto de cálculos é necessário, antes de qualquer outro procedimento, pressionar-se o botão “Analisar”. Assim que este controlo é acionado, surge uma janela informativa que solicita ao utilizador que faça um clique, aproximadamente, no centro de uma das bactérias observadas na imagem. Depois de a bactéria pretendida ter sido selecionada, as coordenadas da imagem associadas a esse evento ( $m, n$ ) são guardadas e em redor dessa localização é definida

uma região quadrangular (de dimensões  $15 \times 15$ ), a qual será utilizada para o cálculo representativo da média e da mediana dos valores de intensidade associados às bactérias. Estes dois parâmetros estatísticos são obtidos através da aplicação de duas funções do MATLAB: a função `mean` (para o cálculo do valor médio) e a função `median` (para o cálculo do valor mediano). Por último, estes valores são projetados na coluna de elementos localizada mais à direita no painel, a qual diz respeito à análise estatística das *E. coli*. De seguida, uma nova janela informativa emerge, a qual exige ao utilizador a seleção de uma área do *background* para análise. Recorrendo à função `imcrop` do MATLAB, é possível definir-se interactivamente, com o auxílio do rato, o tamanho e a posição deste elemento. Depois do utilizador ter selecionado a região de interesse, as restantes etapas do processo seguem exatamente o mesmo protocolo adotado para o caso das bactérias. Imediatamente após o valor da média e da mediana terem sido calculados, os seus resultados são projetados na coluna de elementos localizada mais à esquerda no painel.

O terceiro elemento, pertencente ao painel da “Segmentação”, como o próprio nome sugere, permite aplicar o algoritmo de segmentação desenvolvido no âmbito desta tese. Este tópico irá ser descrito, de um modo pormenorizado, no subcapítulo subsequente.

### 3.3 Algoritmo

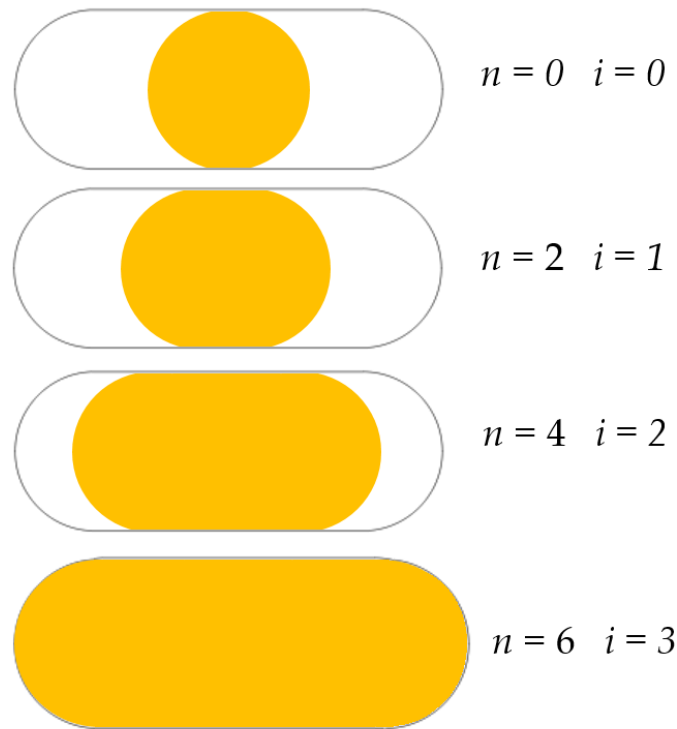
O algoritmo de segmentação desenvolvido no âmbito desta tese pode ser separado em 7 etapas distintas, as quais irão ser abordadas, de um modo sequencial e pormenorizado, ao longo deste subcapítulo. À medida que o protocolo desenvolvido for sendo explicado, resultados ilustrativos de cada transformação aplicada à imagem vão sendo apresentados.

Um dos principais objetivos que impulsionou o desenvolvimento deste algoritmo foi transformar o processo de segmentação das *Escherichia coli* (contidas nas imagens providenciadas pelo LBD) numa abordagem simples, rápida, intuitiva e que exigisse o mínimo de interação possível por parte do utilizador.

Atualmente, o método que tem sido empregue para este efeito requer inicialmente, para cada bactéria, a execução de dois cliques (um em cada extremidade da *E. coli*). Após a execução desta tarefa, surge no plano da imagem um conjunto de  $n$  pontos ajustáveis que se distribuem uniformemente ao longo do segmento de reta que une os dois extremos de cada bactéria. De seguida, cada ponto é individualmente ajustado na imagem de maneira a que, no final, cada conjunto de pontos possa assumir a forma da bactéria à qual diz respeito. Considerando que uma imagem pode possuir dezenas ou até mesmo centenas de bactérias, este método torna-se facilmente demasiado exaustivo e demorado.

Com o algoritmo proposto nesta dissertação, procurou-se reduzir o número de cliques por bactéria para apenas um e, a partir daí, tornar o processo de segmentação tão automático quanto possível.

Este algoritmo baseia-se na aproximação da morfologia de cada bactéria a uma série de círculos de raio  $r$ , correspondendo  $r$  à largura (ou diâmetro) da *E. coli* considerada (ver **Figura 3.10**). Inicialmente, é definido um primeiro círculo que, através de um conjunto de condições, se ajusta à dimensão da largura do microrganismo ao qual diz respeito. Este primeiro círculo, de modo a ser diferenciado dos restantes, passará a ser denominado por círculo central (por se posicionar, aproximadamente, no centro da bactéria). A partir deste elemento inicializador serão gerados  $n$  círculos que se propagam em direções opostas na bactéria. Considerando, como exemplo, uma *E. coli* disposta horizontalmente no plano da imagem, isto implicaria que, nesta fase inicial,  $\frac{n}{2}$  círculos seriam posicionados à direita do círculo central e  $\frac{n}{2}$  à esquerda.



**Figura 3.10:** Representação gráfica do procedimento associado ao algoritmo de segmentação proposto.

O presente algoritmo foi projetado para que a cada iteração,  $i$ , fossem gerados dois círculos centralizados em pontos da imagem progressivamente mais distantes do círculo central. Assim, a cada iteração, o número de círculos ( $N$ ) que preenchem o interior da bactéria é obtido através da seguinte expressão:

$$N = 2^i + 1$$

onde

$$i = 0, 1, \dots, I$$

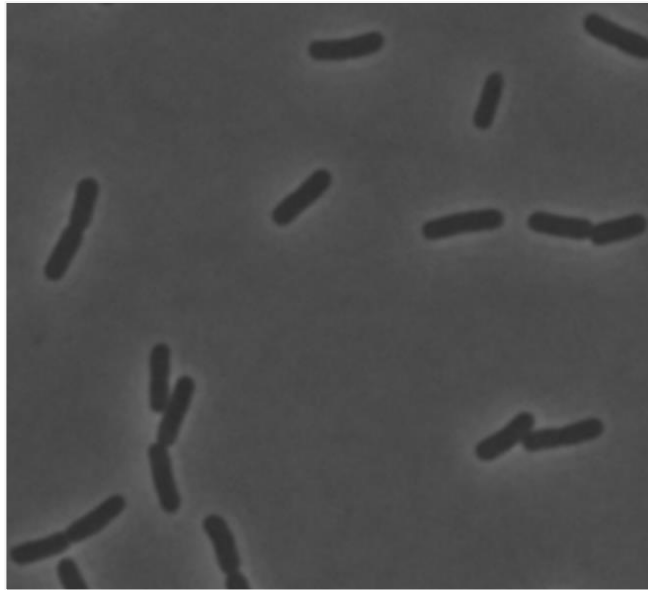
Caso o círculo central, inicialmente, não tenha sido corretamente posicionado, isto poderá implicar que, numa dada iteração,  $j$ , um dos dois círculos gerados atinja uma das extremidades da bactéria sem que o outro tenha detetado a extremidade oposta. A partir desse momento, apenas o círculo que não atingiu nenhuma extremidade será considerado para continuar a preencher a bactéria e neste caso o número total de círculos passará a ser calculado, até que a bactéria esteja completa, através da seguinte equação:

$$N = 2^j + 1 + i$$

com  $j = 0, 1, \dots, J$  e  $i = 0, 1, \dots, I$ .

### 3.3.1 Imagens

As imagens que serviram de objeto de estudo para o desenvolvimento do presente algoritmo, tal como mencionado anteriormente, são imagens obtidas por microscopia de contraste de fase e providenciadas pelo LBD especificamente para este projeto. Distribuídas por este conjunto de imagens monocromáticas (com  $2560 \times 1920$  píxeis a 8 bits), encontram-se diversas populações de *Escherichia coli*. Estes microrganismos, tal como se pode verificar na **Figura 3.11**, são representados com uma tonalidade de cinza mais escura do que a tonalidade que caracteriza o plano de fundo da imagem. Esta propriedade é indicativa de que este conjunto de imagens apresentam um contraste de fase positivo.



**Figura 3.11:** Exemplo das imagens disponibilizadas pelo *Laboratory of Biosystem Dynamics*. A imagem destacada na figura representa apenas uma porção, composta por  $681 \times 601$  píxeis, da imagem original.

### 3.3.2 Etapa 1: Filtragem

O pré-processamento de imagem é uma das etapas que mais influência exerce nos resultados de segmentação obtidos. Esta etapa traduz-se, frequentemente, na aplicação de um conjunto de métodos que atuam de modo a realçar determinadas características na imagem (como por exemplo, os contornos de um dado objeto de interesse) ou de modo a atenuar eventuais ruídos que se sobreponham à informação que irá ser posteriormente processada.

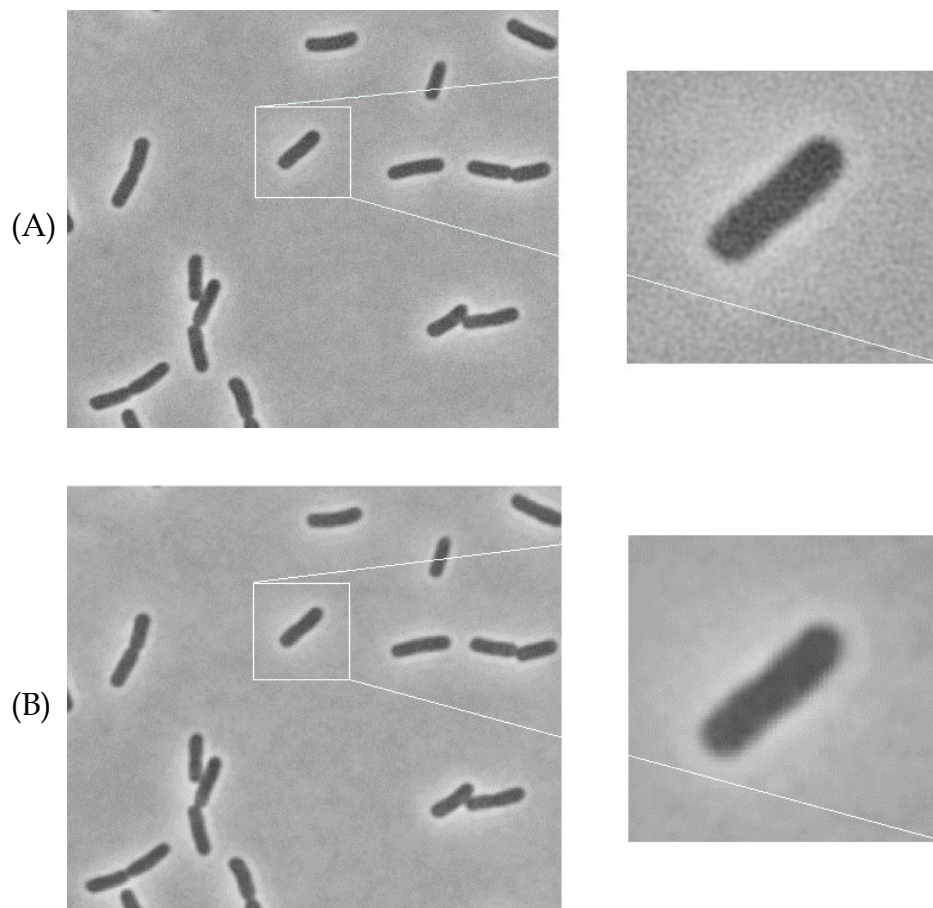
No caso concreto das imagens fornecidas pelo LBD, foi possível identificar-se, logo à partida, uma característica considerada indesejável: a contaminação das imagens com ruído do tipo Gaussiano.

O ruído Gaussiano é um tipo de ruído habitualmente originado pelos sistemas de aquisição de imagem digital. Algumas das causas mais comuns associadas ao aparecimento desta contaminação são nomeadamente uma má iluminação, temperaturas elevadas ou problemas na transmissão de dados. Através do pré-processamento de imagens, é possível atenuar parcialmente os efeitos desta interferência na imagem. No entanto, esta atenuação provoca, na maioria dos casos, a perda de alguns detalhes presentes na imagem. Algumas das técnicas espaciais que visam suavizar este tipo de ruído são nomeadamente o filtro de média e o filtro Gaussiano [62].

Depois de se avaliar o resultado da aplicação de cada método de suavização (mencionado no parágrafo anterior) ao conjunto de imagens em estudo, verificou-se que aquele que melhores resultados produzia era o filtro de média de dimensões  $5 \times 5$ . Através desta técnica de filtragem,



foi possível reduzir-se significativamente o efeito do ruído Gaussiano sem se perder, em demasia, a nitidez dos contornos das bactérias. O resultado desta etapa pode ser visualizado na **Figura 3.12**, na qual se encontra realçada uma zona específica da imagem original e a zona correspondente na imagem pré-processada.



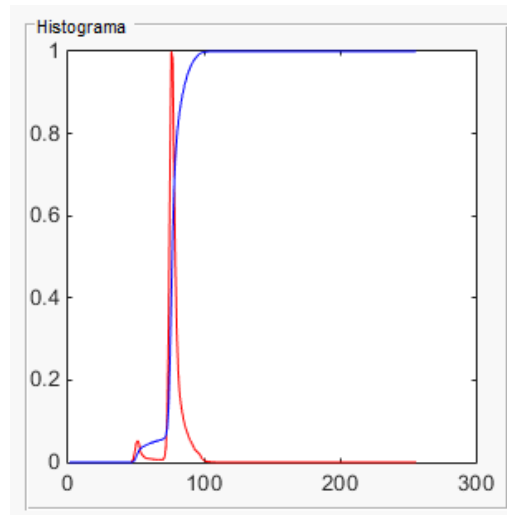
**Figura 3.12:** Exemplo da aplicação do filtro Gaussiano para a atenuação do ruído da imagem. O nível do contraste e brilho de ambas as imagens foi aumentado de modo a facilitar a visualização do resultado da suavização. (A) Imagem original; (B) Imagem suavizada.

### 3.3.3 Etapa 2: Extração de contornos

De modo a aplicar-se o algoritmo de segmentação desenvolvido, foi necessário, ainda associado à etapa do pré-processamento, extrair-se o conjunto de contornos que definem a morfologia, posição e distribuição das bactérias na imagem. Com este objetivo em mente, desenvolveu-se um procedimento computacional bastante simples composto por três métodos estrategicamente adotados.

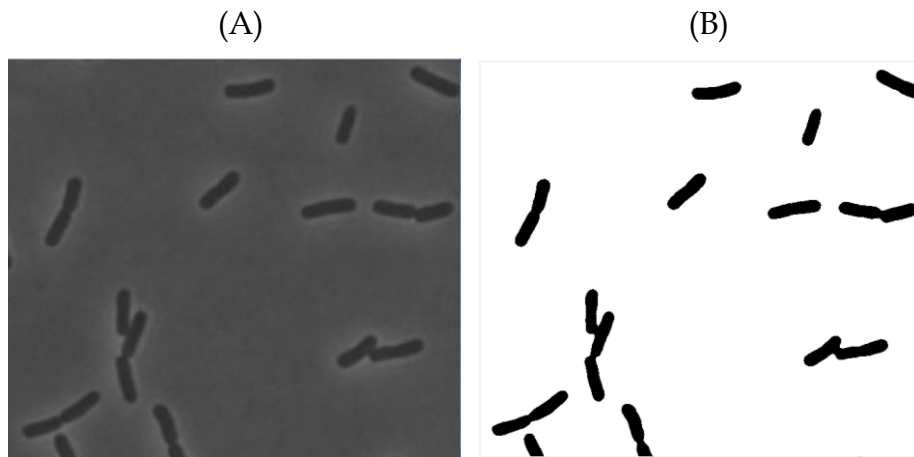
Através da aplicação do primeiro e segundo método, foi possível obter-se uma binarização da imagem previamente filtrada. A binarização consiste no processo através do qual uma imagem, definida inicialmente em tons de cinza, é convertida numa representação binária isto é, numa imagem composta por apenas por dois níveis de intensidade (tipicamente, o branco e o preto). Esta técnica, por si só, é frequentemente empregue enquanto algoritmo de segmentação no entanto, quando aplicada ao conjunto de imagens de estudo deste projeto, os seus resultados revelaram-se insuficientes e por isso a binarização foi introduzida apenas como mais uma etapa do algoritmo de segmentação proposto [63]. O principal objetivo associado a esta técnica consiste em separar os diversos objetos contidos na imagem do plano de fundo. Para tal, é necessário definir-se um limiar de intensidade (também denominado por *threshold*), o qual irá ser comparado com o valor de cada píxel pertencente à imagem original. Caso o valor de intensidade assumido por um dado píxel seja inferior ao valor previamente estabelecido para o *threshold*, o píxel em questão passará a ser representado com a cor preta; caso contrário, o píxel passará a ser representado pela cor branca. Atualmente existem inúmeros métodos que podem ser utilizados para determinar o valor ideal a atribuir ao *threshold*. Uma das técnicas mais frequentemente aplicadas para determinar um limiar ótimo de intensidade é o método de Otsu. Este algoritmo revela-se bastante eficiente nos casos em que a imagem original é representada por um histograma do tipo bimodal (isto é, um histograma onde podem ser observados apenas dois picos). O princípio de funcionamento associado a este método consiste em aproximar o histograma da imagem a duas funções Gaussianas e definir um valor de *threshold* que minimize a variância intra-classes [64].

Ao analisar-se, a título de exemplo, o histograma da imagem filtrada destacada na **Figura 3.12 (B)**, verificou-se que esta representação se tratava de um histograma do tipo bimodal. Assim sendo, como primeira etapa do processo de binarização, aplicou-se o comando `graythresh` à imagem de entrada. Através desta função, baseada no método de Otsu, foi possível determinar-se o valor de *threshold* ótimo para a executar a binarização da imagem. O histograma considerado neste exemplo encontra-se destacado a vermelho na **Figura 3.13**. A azul, nesta mesma figura, pode ser visualizado o histograma acumulado da imagem considerada.



**Figura 3.13:** Histograma e histograma acumulado da imagem presente na Figura 3.11.

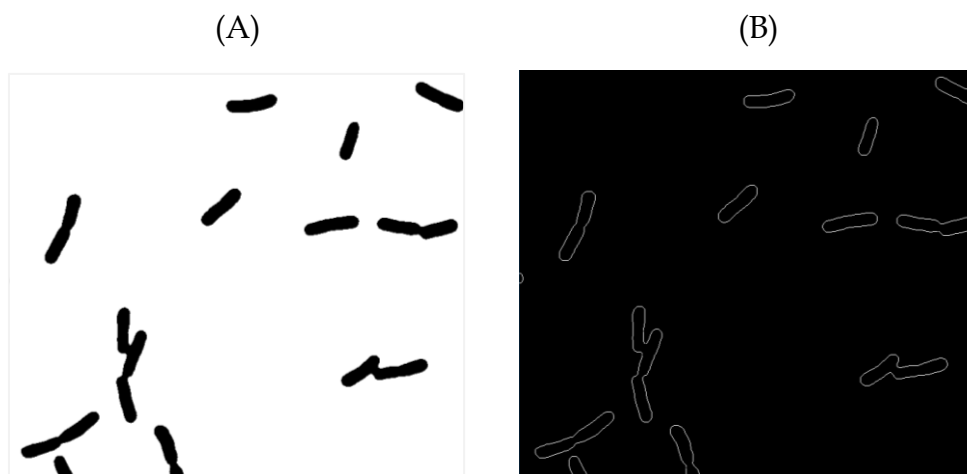
Depois do valor ideal para o *threshold* ter sido calculado, esse valor é seguidamente utilizado para converter a imagem filtrada numa imagem binária. Esta transformação é implementada através da função *imbinarize*, a qual opera substituindo todos os valores da imagem acima de um determinado valor de *threshold* por 1 e os restantes por 0. O resultado da aplicação destes dois primeiros métodos é realçado na **Figura 3.14**, abaixo destacada.



**Figura 3.14:** Exemplo da aplicação do processo de binarização baseado no método de Otsu à imagem suavizada. (A) Imagem suavizada; (B) Imagem binarizada.

Por último, de modo a extrair-se os contornos das bactérias, foi necessário submeter-se a imagem binarizada a um processo de filtragem de detecção de bordas. O filtro selecionado para este efeito foi o filtro Laplaciano, o qual pode ser computacionalmente construído através da função *fspecial* do MATLAB e aplicado à imagem através do comando *imfilter*. A convolução

da imagem binarizada com este operador matricial produz uma nova imagem constituída apenas pelas transições de intensidade e pequenos detalhes. O resultado desta filtragem pode ser visualizado na **Figura 3.15**, abaixo apresentada.



**Figura 3.15: Exemplo da aplicação do processo do operador Laplaciano à imagem binarizada.** (A) Imagem binarizada; (B) Imagem obtida através da aplicação do filtro de Laplace.

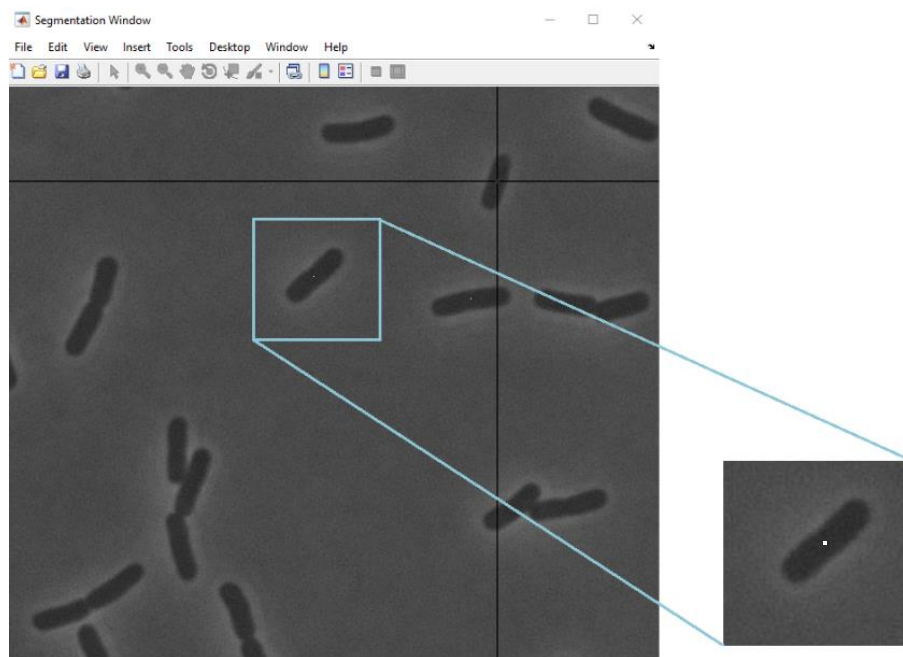
### 3.3.4 Etapa 3: Seleção de bactérias

Após a extração dos contornos que delimitam as bactérias distribuídas na imagem de entrada, é gerada pelo programa uma janela informativa que solicita ao utilizador que selecione, por intermédio de uma série de cliques, o conjunto de bactérias que pretende segmentar. É especificado, na informação contida nesta janela, que cada clique deve ser efetuado, aproximadamente, no centro de cada bactéria-alvo.

Depois de se pressionar o botão “OK” associado a esta janela informativa, o processo de seleção de bactérias é inicializado e, imediatamente depois, surge na imagem um sistema de eixos móvel. Este elemento, introduzido na figura através da função `ginput(n)`, permite ao utilizador identificar  $n$  pontos na imagem, movimentado o cursor com o auxílio do rato. A cada clique efetuado, este comando retorna as coordenadas do local da imagem onde ocorreu esse evento.

Este processo de seleção foi implementado de modo a que cada bactéria, à exceção da última, possa ser iterativamente selecionada pelo utilizador com o auxílio do botão esquerdo do rato. A última bactéria, por sua vez, deverá ser identificada com o botão direito. Esta distinção visa sinalizar o programa que já não existem mais bactérias a segmentar. Com o objetivo de evitar, por lapso, a ocorrência de dois cliques numa mesma bactéria, implementou-se uma função que converte cada píxel selecionado, à medida que os cliques forem sendo executados, num píxel de

cor branca. Desta forma, tal como demonstrado na **Figura 3.16**, é possível identificar quais as bactérias que já foram selecionadas e as que ainda estão por assinalar.

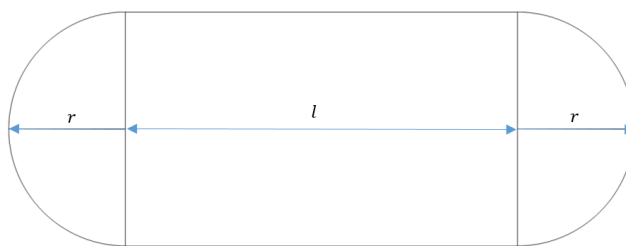


**Figura 3.16:** Exemplo do resultado obtido através da etapa da seleção de bactérias. Na porção da imagem realçada encontra-se destacada uma bactéria previamente selecionada pelo utilizador e agora identificada e distinguida através de um píxel de cor branca.

### 3.3.5 Etapa 4: Ordenação das bactérias selecionadas

À medida que as bactérias vão sendo selecionadas pelo utilizador, uma matriz contendo as coordenadas espaciais dos centros de cada microrganismo assinalado, vai sendo produzida. A partir desta estrutura de dados, devidamente organizada, pretende-se gerar, para cada bactéria assinalada, um círculo central que se ajuste à dimensão da sua largura.

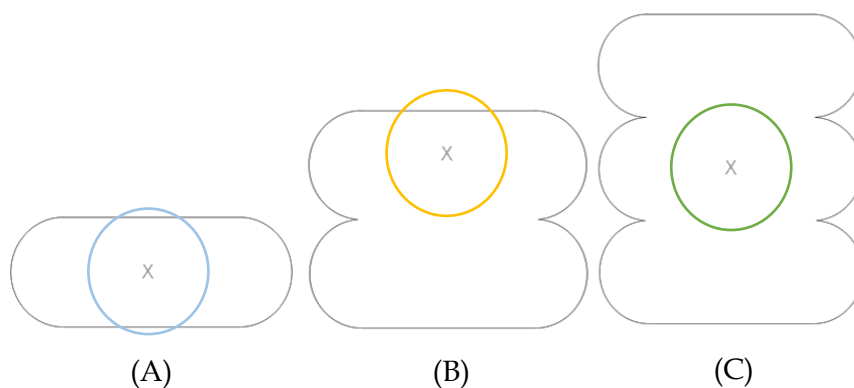
Sabemos que a *Escherichia coli* é uma bactéria bacilar, isto é, uma bactéria cuja morfologia se assemelha à de um bastonete. Este tipo de formato pode ser aproximado a uma estrutura geométrica semelhante à destacada na **Figura 3.17**, a qual é formada por dois semicírculos de raio  $r$  (que definem a largura da bactéria) e um retângulo de comprimento  $l$  (que determina o tamanho do microrganismo).



**Figura 3.17:** Aproximação da morfologia da *E. coli* a um modelo geométrico constituído por um retângulo e dois semicírculos.

Na introdução teórica foi mencionado que a *Escherichia coli* possui um diâmetro aproximadamente igual a 0.5-1.0  $\mu\text{m}$ ; em píxeis, o valor deste parâmetro também não deverá variar significativamente. Depois de se efetuar uma série de medidas, com o auxílio de algumas das ferramentas previamente desenvolvidas e de outras providenciadas pelo MATLAB, calculou-se a média da largura (ou diâmetro) das *E. coli* presentes nas diferentes imagens e obteve-se, para este parâmetro, um resultado aproximadamente igual a 18 píxeis (*default*). Para novos conjuntos de imagem que sejam introduzidos, este parâmetro deverá ser recalculado e, caso se verifique que o valor obtido varia significativamente do valor médio *default*, alguns parâmetros do algoritmo (a partir deste ponto identificados com um “\*”) deverão ser devidamente ajustados. Este valor servirá de referência para inúmeras etapas do algoritmo proposto, inclusivamente a próxima etapa através da qual o tal círculo central de dimensões ajustáveis é formado. Este processo possui uma complexidade e eficiência que, como iremos verificar mais adiante, depende diretamente da vizinhança na qual se insere cada píxel definido pelo utilizador como o centro de uma dada bactéria. Este procedimento será tão mais simples e eficiente sempre que, na vizinhança desse píxel central, sejam detetados píxeis pertencentes ao contorno da bactéria (isto é, píxeis representados pela cor branca). Assim sendo, de modo a averiguar qual o tipo de vizinhança na qual se insere cada centro considerado, desenvolveu-se um método que verifica, para cada bactéria-alvo, o valor de cada píxel incluído ao longo de uma circunferência de raio 11\* (um valor intencionalmente acima do valor médio do raio obtido) definida em redor do ponto selecionado pelo utilizador como centro dessa mesma bactéria.

Através desta etapa, é possível discriminar dois tipos de vizinhança: uma vizinhança na qual são detetados píxeis pertencentes ao contorno das bactérias - ver **Figura 3.18** (A) e (B) - e uma vizinhança na qual apenas são detetados píxeis pertencentes ao plano de fundo da imagem – ver **Figura 3.18** (C).



**Figura 3.18: Tipos de vizinhança detetados na etapa da ordenação das bactérias selecionadas.** (A) Ocorre a interseção da circunferência de vizinhança (de raio 11\*) com duas bordas da bactéria; (B) Ocorre a interseção da circunferência de vizinhança apenas com uma das bordas da bactéria; (C) A circunferência de vizinhança não intersecciona nenhuma das bordas da bactéria selecionada.

Depois de todos os centros terem sido classificados de acordo com a sua vizinhança, a matriz composta pelas coordenadas dos pontos selecionados pelo utilizador é reorganizada de modo a que os centros que possuem uma vizinhança que intersecciona o contorno das bactérias sejam priorizados relativamente aos restantes.

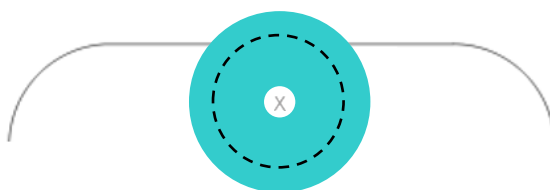
### 3.3.6 Etapa 5: Criação do círculo central

Antes de se desenvolver, para uma dada bactéria selecionada, um círculo central é necessário criar-se uma cor através da qual este microrganismo será, daqui em diante, representado e distinguido. Para efetivar este procedimento, foi criado (para cada *E. coli* identificada) um vetor composto por três elementos. A dimensão e orientação deste vetor bem como os valores assumidos por cada um dos seus elementos pretendem simular os componentes RGB (“Red”, “Green” e “Blue”) que definem a cor dos píxeis das imagens digitais. Depois desta estrutura de dados ter sido criada, é atribuído a cada posição desse vetor um valor aleatório compreendido entre 0 e 255. Sabendo que cada píxel é composto por 3 componentes RGB (um para cada cor) e que cada componente pode assumir 1 de 256 valores de intensidade, através deste modelo é possível gerar-se um total de  $2^{8 \times 3}$  (ou, 16.777.216) cores distintas. Deste conjunto de combinações apenas se excluiu a possibilidade de ser gerada a cor preta (por ser característica do *background*) e a cor branca (por ser característica do contorno das bactérias).

Após a concretização deste passo inicial, fundamental para discernir as diferentes bactérias presentes na imagem, é inicializado o processo que conduz à formação do círculo central.

Este processo é propagado de bactéria para bactéria de acordo com a ordem previamente estabelecida pela matriz de dados resultante da etapa anterior.

Numa primeira fase deste processo, irá ser efetuada uma reavaliação da vizinhança na qual cada centro considerado se insere. Nesta análise, em vez de se observar apenas os píxeis incluídos ao longo de uma circunferência de vizinhança de raio  $11^*$ , consideram-se também outras circunferências de vizinhança com raios compreendidos no seguinte intervalo de valores  $[2; 12^*]$ . Deste modo, a região de vizinhança considerada passa a abranger uma área da imagem semelhante à destacada a cinza na **Figura 3.19**: um anel circular.



**Figura 3.19:** Anel circular que representa a vizinhança considerada para cada bactéria selecionada na etapa da criação do círculo central.

Cada circunferência que compõe este anel será analisada individualmente, começando na de menor raio e terminando na circunferência caracterizada pelo raio superior.

Através da análise dos elementos que compõem as circunferências é possível definir-se, para cada uma delas, quatro tipos de vizinhança:

- **Vizinhança 1:** Sempre que forem detetados, para além dos elementos que constituem o plano de fundo da imagem, um ou mais píxeis de cor branca e nenhum píxel colorido;
- **Vizinhança 2:** Sempre que forem detetados, para além dos elementos que constituem o plano de fundo da imagem, um ou mais píxeis coloridos e nenhum píxel de cor branca;
- **Vizinhança 3:** Sempre que forem detetados, para além dos elementos que constituem o plano de fundo da imagem, píxeis brancos e píxeis coloridos;
- **Vizinhança 4:** Sempre que o conjunto de píxeis detetados for composto apenas por elementos respeitantes ao plano de fundo da imagem;

De seguida, irá descrever o modo como os círculos centrais são formados, consoante o tipo de vizinhança na qual se inserem.

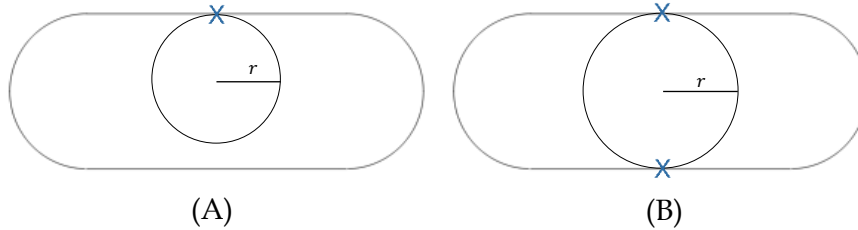


### Vizinhança 1

Uma dada circunferência de vizinhança de raio  $r$  será classificada como uma vizinhança do tipo 1 sempre que este conjunto de pontos intersejar as bordas da bactéria. Esta interseção pode ocorrer em apenas uma das laterais da bactéria (tal como ilustrado na **Figura 3.20 (A)**) ou então, em ambas as laterais (tal como ilustrado na **Figura 3.20 (B)**). Para averiguar em qual das situações se insere a circunferência de vizinhança considerada, calcula-se a distância que separa cada combinação de dois píxeis brancos detetados,  $d$ . Considerando dois pontos genéricos A e B, definidos respetivamente por  $P_A = [x_A; y_A]$  e por  $P_B = [x_B; y_B]$ , este valor é obtido através da seguinte expressão matemática:

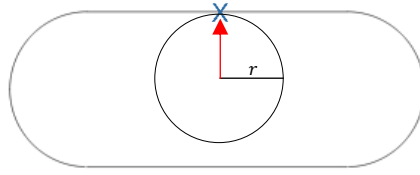
$$d_{AB} = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}$$

Caso se verifique, para um determinado par de píxeis brancos, que a distância que os separa é igual ou superior a  $2 \times r - 1$  píxeis, conclui-se que foram detetados píxeis pertencentes a ambas as laterais da bactéria. Para este caso em particular, em redor do ponto indicado pelo utilizador como centro da bactéria, é gerado um círculo de raio  $r$ .



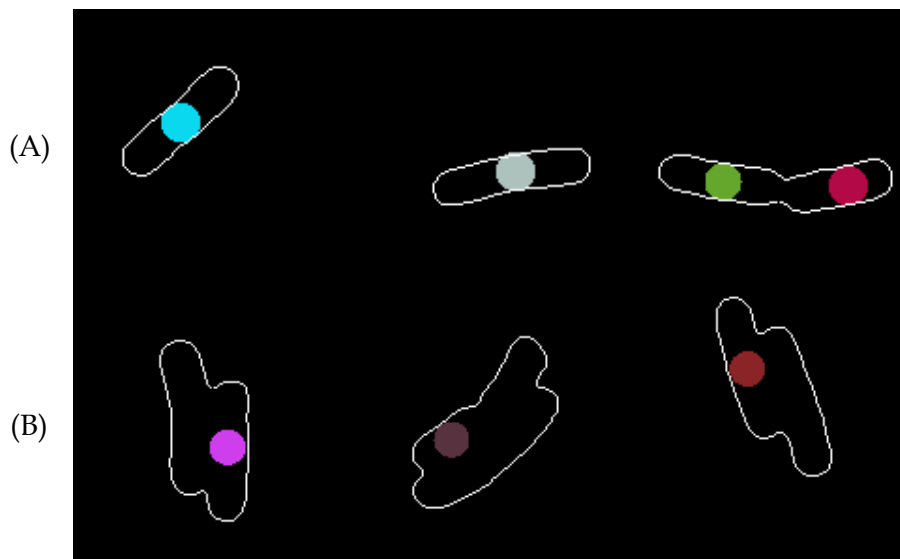
**Figura 3.20:** (A) Interseção da circunferência de vizinhança com uma das bordas da bactéria; (B) interseção da circunferência de vizinhança com ambas as laterais da bactéria considerada.

Caso não se detete, para a circunferência considerada, nenhum par de píxeis brancos separados por este espaçamento ou só se detete um único píxel branco, então o centro da bactéria irá ser transladado na imagem. Este movimento irá ocorrer no sentido oposto ao do vetor que une o centro da circunferência e o primeiro píxel branco a ser detetado (representado a vermelho na **Figura 3.21**) até que a distância entre estes dois pontos seja superior a  $r - 1$ . Depois do centro do círculo central ter sido ajustado, o raio da circunferência de vizinhança aumenta.



**Figura 3.21:** Representação do vetor que une o centro da bactéria considerada ao primeiro píxel branco a ser detetado.

Caso se mantenha, para a nova circunferência, o mesmo tipo de vizinhança, todo o procedimento descrito até este ponto volta a ocorrer. O processo termina sempre que forem interseçadas as duas bordas da *E. coli* ou sempre que o raio da circunferência de vizinhança atinja um valor igual a  $10^*$  píxeis. Neste caso, o centro mantém-se estático e gera-se em torno desse ponto um círculo de raio  $9^*$ . Alguns dos resultados obtidos para este tipo de vizinhança podem ser visualizados na **Figura 3.22**. Nesta imagem encontram-se destacados três exemplos para o caso em que as duas bordas são interseçadas (A) e outro três para quando apenas uma das delas é detetada (B).



**Figura 3.22: Resultados exemplificativos da aplicação da etapa da criação do círculo central em bactérias inseridas numa vizinhança inicial do tipo 1.** (A) Exemplos em foram detetadas as duas bordas da bactéria selecionada; (B) Exemplos em que foi detetada apenas uma borda da bactéria selecionada.

## Vizinhança 2

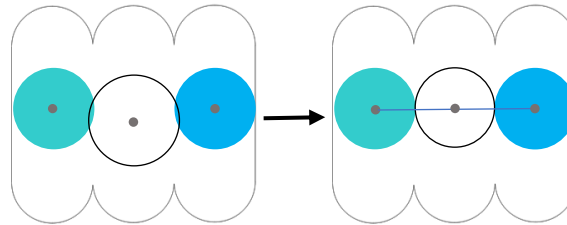
Uma dada circunferência de raio  $r$  é classificada como vizinhança do tipo 2 sempre que intersestar um ou dois círculos respeitantes a outras bactérias. De modo a determinar o número de círculos intersetados, à semelhança do que se efetuou para a vizinhança do tipo 1, calcula-se a distância entre cada combinação de dois píxeis coloridos detetados.

Caso se verifique a existência de um par de píxeis coloridos separados por uma distância igual ou superior a  $2 \times r - 1$  píxeis, conclui-se que houve interseção da circunferência de vizinhança com os círculos inerentes às duas bactérias mais próximas. Assim sendo, irá gerar-se, para a *E. coli* considerada, um círculo centrado no ponto médio do segmento de reta que une os centros dos dois círculos vizinhos mais próximos. O valor do raio associado a este círculo central é obtido através pela seguinte expressão:

$$r = \frac{d_{AB} - r_A - r_B}{2}$$

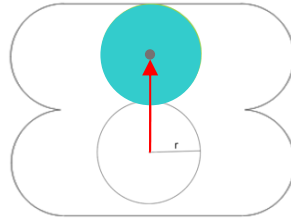
onde  $d_{AB}$  representa a distância que separa os centros dos círculos A e B,  $r_A$  o raio que caracteriza o círculo A e  $r_B$  o raio que caracteriza o círculo B.

Este tipo de ajuste encontra-se representado na **Figura 3.23**, abaixo destacada.



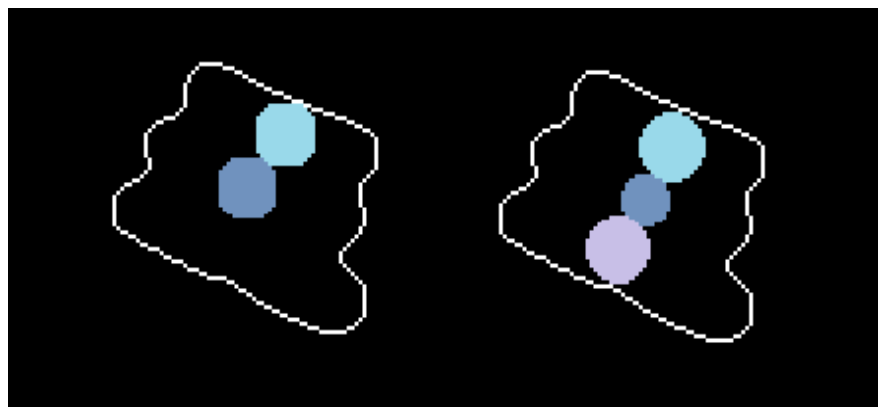
**Figura 3.23:** Representação do tipo de ajuste aplicado sempre que são detetadas duas bactérias a uma distância igual ou superior a  $2 \times r - 1$  píxeis do centro da bactéria considerada.

Caso a condição anterior não se verifique, conclui-se que a circunferência de vizinhança considerada intersetou apenas um único círculo. Neste caso particular, o centro da bactéria será deslocado  $r_C + r - 1$  píxeis na imagem (relativamente ao centro do círculo mais próximo). Na expressão do deslocamento  $r_C$  representa o raio do círculo intersetado pela circunferência de vizinhança de raio  $r$ . Este movimento irá ocorrer no sentido oposto ao do vetor que une o centro da circunferência de vizinhança e o centro do círculo ao qual pertencem os píxeis coloridos detetados (representado a vermelho na **Figura 3.24**).



**Figura 3.24:** Representação do vetor que une o centro da bactéria considerada ao primeiro píxel colorido a ser detetado.

Depois deste deslocamento ter sido efetuado, o raio da circunferência aumenta e, caso se mantenha o mesmo tipo de vizinhança e circunstâncias, este procedimento volta a ser executado até que se intersem dois círculos ou até que o raio da circunferência seja igual a  $10^*$  píxeis. Sempre que esta última situação ocorrer, o centro da bactéria deixa de se movimentar e gera-se, em torno desse ponto, um círculo de raio  $9^*$ . Na **Figura 3.25** encontram-se destacados dois exemplos-tipo dos resultados obtidos através desta metodologia um, para o caso da interseção ter ocorrido com apenas um círculo (A) e outro, para quando são intersetados dois círculos distintos (B).



(A)

(B)

**Figura 3.25: Resultados exemplificativos da aplicação da etapa da criação do círculo central em bactérias inseridas numa vizinhança inicial do tipo 2.** (A) Exemplo dos casos em que a circunferência de vizinhança intersesta apenas um círculo central referente a outra bactéria; (B) Exemplo dos casos em que a circunferência de vizinhança intersesta dois círculos centrais.

### Vizinhança 3

Uma vizinhança do tipo 3 surge sempre que sejam detetados, na mesma circunferência de vizinhança, píxeis brancos e píxeis coloridos. Neste caso, a abordagem a adotar é bastante mais simples do que as abordagens aplicadas aos dois tipos de vizinhança anteriormente apresentados.

Ao serem detetados estes dois tipos de píxeis numa dada circunferência de vizinhança de raio  $r$ , o primeiro passo a executar consiste em calcular para cada par de píxeis (constituído por um píxel branco e outro colorido) a distância que os separa. Depois de esta operação ter sido aplicada a todas as combinações possíveis, é gerado um conjunto de resultados organizados numa matriz. A partir desta estrutura de dados determina-se o par de píxeis que se encontra separado por uma menor distância.

Depois deste par ter sido detetado, segue-se a etapa de formação do círculo central. Este elemento irá centralizar-se no ponto médio do segmento de reta que une os dois píxeis do par considerado e irá ser caracterizado por um raio igual a metade da distância que separa estes dois pontos. Dois resultados exemplificativos da aplicação deste ajuste são apresentados na **Figura 3.26**.



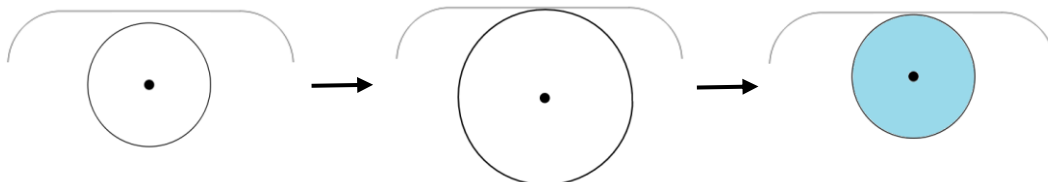
**Figura 3.26:** Resultados exemplificativos da aplicação da etapa da criação do círculo central em bactérias inseridas numa vizinhança inicial do tipo 3.

### Vizinhança 4

Uma vizinhança do tipo 4 consiste numa circunferência de raio 9 que não detetou nenhum píxel colorido nem branco. Para este caso específico, continuam a definir-se vizinhanças até se atinja o limite máximo: um raio igual a  $12^*$  píxeis. Durante este processo, caso se detete algum ponto não pertencente ao plano de fundo da imagem, as coordenadas desse elemento são

guardadas e cria-se um círculo de raio  $9^*$  adjacente ao píxel detetado, tal como representado na **Figura 3.27**.

Nos casos mais extremos em que seja projetada uma circunferência de vizinhança de raio  $12^*$  e mesmo assim não se detete nenhum píxel colorido nem branco, gera-se um círculo de raio  $9^*$  centrado no local inicialmente seleccionado pelo utilizador.

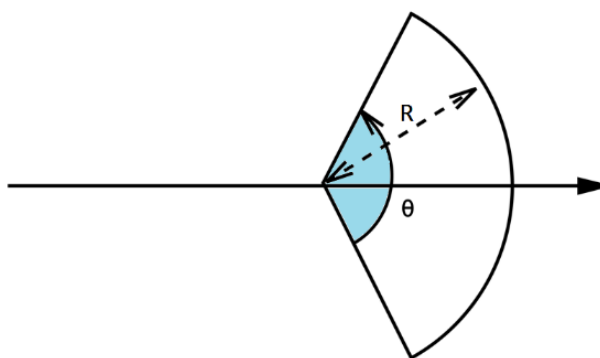


**Figura 3.27:** Representação gráfica do tipo de ajuste a aplicar a bactérias que, na etapa da criação do círculo central, se inseriram numa vizinhança do tipo 4.

### 3.3.7 Etapa 6: Setor circular direcional

Depois de todos os círculos centrais terem sido gerados, inicia-se, para cada bactéria, uma nova etapa do algoritmo: a etapa que conduz à formação de um par de setores circulares denominados por direcionais.

Um setor circular constitui numa porção do círculo limitada por dois raios e um arco. Esta figura geométrica é definida por dois parâmetros: a medida do raio,  $R$ , e a dimensão do ângulo central,  $\theta$  (ver **Figura 3.28**).



**Figura 3.28:** Setor circular de raio  $R$  e ângulo de abertura  $\theta$  (adaptado de [65]).

Através desta etapa do algoritmo, para cada bactéria seleccionada, irão ser gerados dois setores circulares direcionais. Estes par de elementos são simétricos e encontram-se centralizados no mesmo ponto: o centro do círculo central gerado na etapa anterior.

Tal como mencionado anteriormente, o princípio de funcionamento do algoritmo proposto baseia-se na propagação de uma série de círculos de ao longo da bactéria, começando no centro e expandindo-se para cada uma das suas extremidades. Para tal, é necessário definir-se a direção da propagação destas figuras geométricas. Este parâmetro é calculado com o auxílio destes setores circulares direcionais, tal como iremos verificar mais à frente neste tópico.

Para se formar o setor direcional é necessário, para cada *Escheria coli*, aceder-se a dois conjuntos de valores: as coordenadas do centro e a dimensão do raio que definem o seu círculo central.

A partir do raio deste círculo,  $r$ , define-se o valor do raio do setor circular,  $R$ , o qual é calculado através da seguinte igualdade:

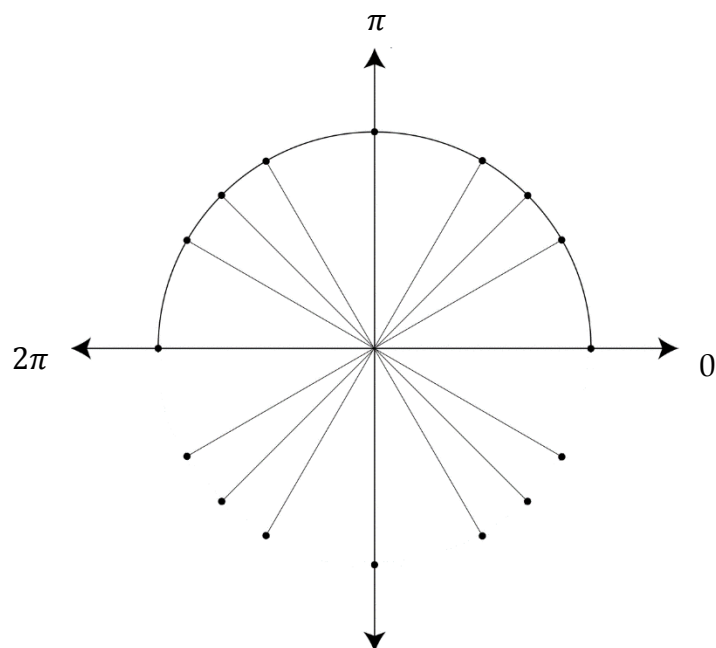
$$R = \frac{4}{3}r$$

Depois de ter sido definido, para cada bactéria, o valor de  $R$ , é necessário calcular-se o valor a atribuir ao ângulo  $\theta$ . É o valor deste parâmetro que irá determinar a direção de propagação da bactéria, o qual é calculado através do seguinte método.

Considerando como referência o centro do círculo central da bactéria irão definir-se  $n$  segmentos de retas cujo ponto médio, para todos eles, consiste precisamente nesse ponto de referência. Estes segmentos possuem uma dimensão igual a  $2R$  e podem assumir qualquer orientação compreendida no intervalo de  $[0; \pi]$  relativamente ao eixo das abcissas da imagem,  $f(m,n)$  (ver **Figura 3.29**).

Cada segmento de reta será considerado apenas se nenhum dos elementos que o constitui, nem numa vizinhança de  $3 \times 3$  de cada elemento, for detetado um píxel branco ou um píxel pertencente a outra bactéria. Primeiramente irá avaliar-se a hipótese de se definir um segmento para 0 radianos, seguidamente esta hipótese será avaliada para  $\frac{\pi}{60}$  e assim sucessivamente até que se atinja  $\pi$  radianos.

À medida que este conjunto de segmentos for sido definido e contabilizado, é calculado, paralelamente, o número de setores circulares gerados por intermédio deste método (o qual, idealmente, deverá ser igual a dois).



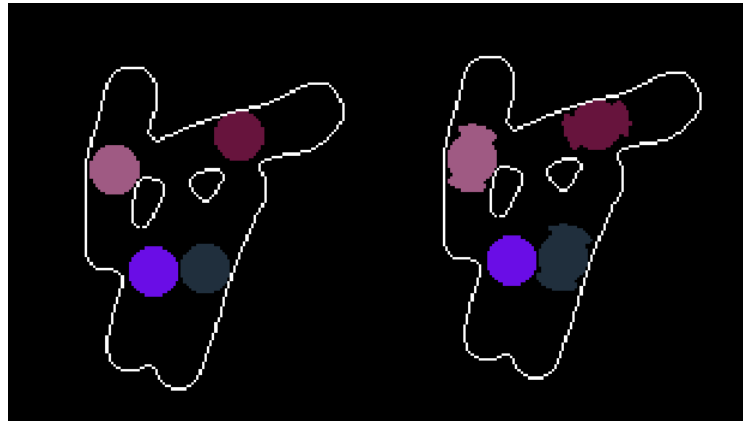
**Figura 3.29: Ilustração do processo a partir do qual se forma o par de setores circulares.** O centro do sistema de eixos da figura representa o ponto da imagem definido como o centro de uma dada bactéria considerada. O raio do semicírculo que se forma entre os 0 e os  $2\pi$  radianos (relativamente ao eixo das abcissas da imagem) é igual a  $R$ .  $[0;2\pi]$  representa o intervalo de ângulos para o qual será testada a hipótese da formação de cada reta que poderá compor o par de setores circulares (adaptado de [66]).

Sempre que forem originados mais dois setores circulares direcionais, conclui-se que, relativamente ao caso ideal, a direção de propagação desta bactéria não se encontra tão bem definida. Assim que detetados, estes casos serão separados dos restantes, colocados numa matriz (denominada por “undefinedNewCenters”) com toda a informação considerada relevante e avaliados numa fase posterior do algoritmo. Os casos ideais, por sua vez, são colocados numa matriz distinta (denominada por “definedNewCenters”) a qual deverá ser ordenada da bactéria com a direção melhor definida para a bactéria com a direção menos bem definida. Para as bactérias incluídas na matriz undefinedNewCenters, os setores gerados no processo, tal como ilustrado na **Figura 3.30**, são eliminados de modo a não interferir com as restantes etapas do processo.

Para se definir esta ordem calcula-se, para cada caso geral, o ângulo que caracteriza a abertura dos dois setores definidos,  $\theta$ . Quando menor for esse ângulo, mais bem definida estará a direção da bactéria considerada. Para além de  $\theta$ , calcula-se ainda um segundo ângulo,  $\alpha$ , que define a orientação da bactéria no plano da imagem.

Por último, de modo a definir-se quais os centros dos próximos dois círculos determina-se a média dos píxeis que ocupam o arco de cada setor circular gerado.





**Figura 3.30: Exemplo do resultado da aplicação da etapa da formação dos setores circulares direcionais.** Para três das quatro bactérias selecionadas neste exemplo foi possível definir-se a direção de propagação dos círculos na bactéria. No caso da bactéria representada com a cor violeta, a direção de propagação não ficou bem definida e por isso esta será processada numa fase posterior relativamente às restantes.

### 3.3.8 Etapa 7: Propagação

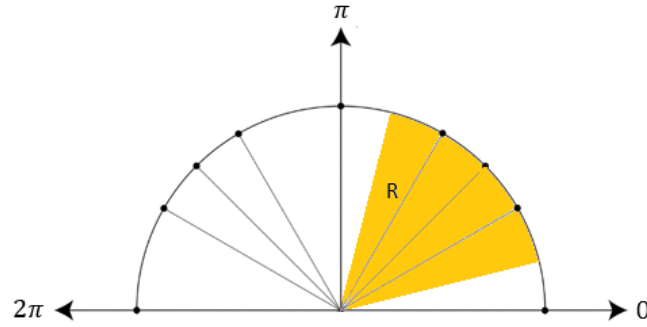
A etapa da propagação é um processo ciclico que se inicia criação dos dois círculos cujos centros foram calculados através etapa anterior. Imediatamente antes deste par de círculos ser formado é necessário averiguar-se a vizinhança de cada um deles. Nesta etapa, considera-se como vizinhança uma circunferência definida exatamente pelo mesmo raio que o do círculo central da bactéria. De acordo com o tipo de vizinhança detetada, caso seja necessário, procede-se a um ajuste da posição do círculo de acordo um conjunto de condições muito próximas às definidas na secção 3.3.6. A única diferença significativa é que nesta etapa, para a maioria dos casos, o raio da circunferência de vizinhança mantêm-se inalterado. A única exceção a esta regra são os casos em que na circunferência de vizinhança não são detetados píxeis pertencentes a outras bactérias (isto é, píxeis coloridos) nem píxeis associados ao contorno da bactéria considerada (isto é, píxeis brancos). Neste caso particular, a circunferência de vizinhança irá aumentar progressivamente, de um em um píxel, até atingir um raio igual a  $r + 4$  ( $r$  representa, nesta expressão, o valor associado ao raio do círculo central da bactéria a ser propagada). Assim que for detetado um píxel branco ou colorido a circunferência deixa de se expandir e, para o conjunto de pontos detetados, averigua-se qual o píxel que se encontra mais próximo do centro calculado. Depois deste elemento ter sido determinado, o centro do círculo será deslocado no sentido oposto ao do vetor que une o centro e a localização deste píxel. Esta translação irá ocorrer até que o centro do círculo se distancie  $r + 1$  píxeis do ponto detetado.

De seguida, inicia-se (para cada círculo agora ajustado) a etapa da formação dos setores circulares direcionais. Nesta fase, a dimensão do raio que define cada um dos setores diminui,

comparativamente ao raio estipulado anteriormente. Nesta fase do algoritmo este parâmetro passa a ser determinado pela seguinte expressão:

$$R = \frac{6}{5}r$$

Outra alteração crucial que surge neste procedimento, relativamente ao anterior, relaciona-se com o intervalo de ângulos que é considerado para se definir as retas de comprimento  $2R$  que constituem os setores circulares. Nesta etapa, de modo a evitar que a propagação dos círculos ocorra para zonas da imagem correspondentes a outras bactérias, estipulou-se que se procuraria definir estas retas direcionais dentro do mesmo intervalo de ângulos que caracteriza o par de setores circulares definido para o primeiro círculo da bactéria a ser formado. Assim, se para esse primeiro círculo o ângulo de abertura for, por exemplo, igual a  $\frac{\pi}{3}$  e no intervalo considerado de  $[0; \pi]$  o primeiro raio do setor for obtido a  $\frac{\pi}{3}$  e o segundo a  $\frac{2\pi}{3}$ , durante a fase da propagação só se irá procurar definir retas direcionais nesse mesmo intervalo:  $[\frac{\pi}{3}; \frac{2\pi}{3}]$ . Esta restrição encontra-se ilustrada na **Figura 3.31**.



**Figura 3.31:** Restrição do intervalo de ângulos para o qual será testada a hipótese da formação de cada reta que poderá compor o próximo par de setores circulares direcionais (adaptado de [66]).

Depois de se ter formado, para cada círculo, um par de setores circulares, calcula-se o centro do próximos dois círculos através da média dos píxeis incluídos nos arcos dos dois setores circulares que se encontrem mais distantes do círculo central.

O procedimento descrito até agora volta a repetir-se até que para um dado círculo sejam detetados mais do que 1 par de setores (isto é, até que a direção de propagação da bactéria não esteja bem definida) ou que o número de retas que constituem o par de setores seja inferior a determinado limiar previamente estabelecido (esta condição indica que pelo menos um dos círculos formados atingiu uma das extremidades da bactéria).

A etapa da propagação, numa fase inicial, é apenas aplicada às bactérias incluídas no vetor `definedNewCenters`. Relativamente às *E. coli* inseridas no vetor `undefinedNewCenters`, essas irão ser primeiramente re-submetidas à etapa da formação dos setores circulares (desta vez para um valor de  $R$  ligeiramente superior:  $\frac{3}{2}r$ ) e só depois de concluída essa etapa é que irão ser propagadas. Esta propagação ocorre em duas fases distintas: primeiro propagam-se as bactérias que possuírem uma direção de propagação bem definida e posteriormente as restantes.

Depois de todas as bactérias terem sido propagadas e por um dos dois motivos acima mencionados essa propagação tiver sido suspensa, volta a aplicar-se esta etapa, uma segunda vez, a todas as *E. coli*. No entanto, nesta iteração, estabelece-se um valor de  $R$  inferior (nomeadamente igual a  $\frac{9}{7}r$ ). Esta alteração é feita de modo a garantir que a eventual formação de mais do que um par de setores na iteração anterior não se encontra relacionada com o facto do valor de  $R$  ser demasiado grande e para assegurar que todos os círculos que eventualmente, conforme a condição anteriormente referida, tenham atingido um dos extremos da bactéria, se possam ajustar ainda mais a esse limite.

Por último, são detetados todos os círculos não tenham atingido, através da propagação anterior, um extremo da bactéria. Depois deste conjunto de elementos ter sido identificado, cada círculo passa a ser considerado individualmente e é posteriormente sujeito a uma última propagação, desta vez caracterizada por um  $R$  igual  $\frac{10}{9}r$ .

Na **Figura 3.32**, abaixo destacada, é apresentado um exemplo do resultado da aplicação da etapa da propagação a uma bactéria isolada na imagem.



**Figura 3.32:** Exemplo do resultado da aplicação da etapa da propagação de círculos.

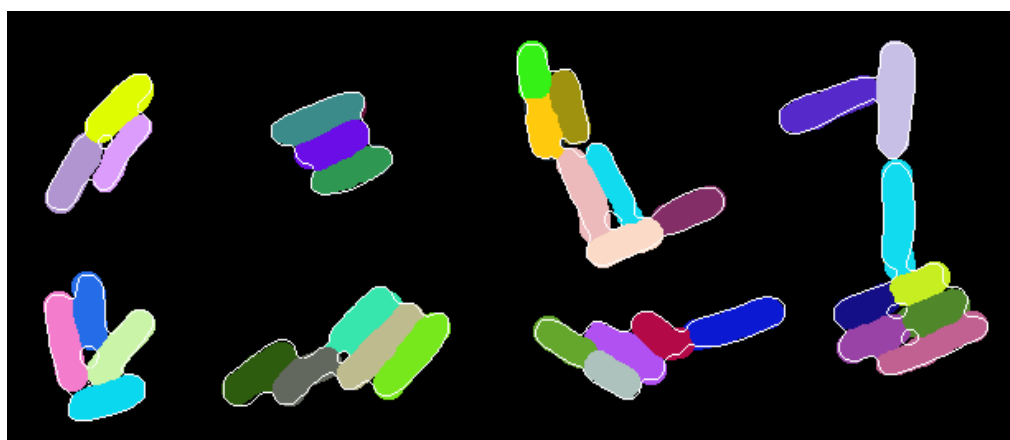


## Análise de Resultados

### 4.1 Resultados obtidos

De modo a avaliar a eficiência do algoritmo implementado através desta dissertação, a aplicação desenvolvida foi testada num conjunto diversificado de imagens. Procurou incluir-se, nesta coletânea, exemplos onde as bactérias se encontrassem isoladas, organizadas em pares e noutra tipo de agrupamentos de maior dimensão. Esta diversidade foi introduzida de modo a poder comparar os resultados obtidos para os diferentes níveis de complexidade das agregações formadas por estes microrganismos.

Alguns resultados, associados a etapas intermédias do procedimento adotado, foram sendo apresentados ao longo do capítulo 3. Na **Figura 4.1**, encontram-se destacados sete resultados finais que demonstram visualmente o efeito da aplicação do algoritmo quando aplicado a diferentes circunstâncias.



**Figura 4.1:** Exemplos do resultado da aplicação do algoritmo de segmentação proposto a diversos conjuntos de bactérias.

## 4.2 Validação do algoritmo

Para efetuar a validação do método de segmentação desenvolvido foi necessário comparar-se o resultado obtido através deste algoritmo com o resultado gerado pela segmentação manual das imagens. A segmentação manual - embora sendo altamente subjetiva - é encarada, nesta fase, como o modelo correto de segmentação. Os critérios utilizados para a validação do algoritmo de segmentação proposto foram a precisão, a sensibilidade e a *F-measure* (ou *F-score*) da classificação.

### Precisão

A precisão (também denominado por taxa de detecção ou valor preditivo positivo) é um critério que permite medir a qualidade da classificação do modelo de segmentação proposto. Este parâmetro é descrito matematicamente através da seguinte expressão:

$$Precisão = \frac{TP}{TP + FP}$$

onde,

TP (do inglês, *true positive*) corresponde ao número de píxeis corretamente classificados como pertencentes a uma dada bactéria (ou conjunto de bactérias) da imagem.

FP (do inglês, *false positive*) corresponde ao número de píxeis erroneamente classificados como pertencentes a uma dada bactéria (ou conjunto de bactérias) da imagem.

O denominador da expressão da precisão,  $TP + FP$ , corresponde ao número total de píxeis classificados pelo algoritmo proposto como pertencentes a uma dada bactéria. Este parâmetro, a partir deste ponto denominado por  $S_A$ , é obtido através da contabilização do número de píxeis coloridos presentes nas imagens resultantes da aplicação do algoritmo a ser validado. O valor associado a  $TP$ , por sua vez, é obtido através da seguinte igualdade:

$$TP = S_A - FP$$

O valor dos falsos positivos (FP), erros do tipo I, é obtido através da subtração da imagem resultante da aplicação do método de segmentação desenvolvido pelo resultado da segmentação manual da imagem.

## Sensibilidade

A sensibilidade (também denominada por revocação ou taxa de verdadeiros positivos) permite calcular a percentagem de verdadeiros positivos detetados, a qual pode ser descrita matematicamente através da seguinte expressão:

$$Sensibilidade = \frac{TP}{TP + FN}$$

onde,

FN (do inglês, *false negative*) corresponde ao número de píxeis erroneamente classificados como pertencentes ao *background* da imagem;

O cálculo dos falsos negativos (FN), erros do tipo II, é obtido através da subtração da imagem segmentada manualmente pela imagem resultante da aplicação do algoritmo proposto.

## *F-measure*

Na análise estatística de uma classificação binária, a *F-measure* é um critério que permite medir a eficiência do método aplicado. Na expressão utilizada para o cálculo do valor deste parâmetro são considerados os valores da precisão e sensibilidade do classificador. A precisão, como vimos anteriormente, é o número de verdadeiros positivos dividido pelo número total de resultados positivos e a sensibilidade, o número de resultados verdadeiros positivos dividido pelo número de verdadeiros positivos que deveriam ter sido detetados.

O valor do parâmetro *F-measure* é obtido através da média harmónica entre a precisão e a sensibilidade:

$$F - measure = 2 \cdot \frac{Precisão \cdot Sensibilidade}{Precisão + Sensibilidade}$$

A **Tabela 4.1**, apresentada em seguida, discrimina os valores de  $S_A$ ,  $S_M$  (número de píxeis classificados pela segmentação manual como pertencentes a uma dada bactéria), FN, FP, TP, precisão (P), sensibilidade (S) e *F-measure* (*F-m*) calculados para 50 imagens distintas. As primeiras 10, representadas a laranja, consistem em porções de imagens onde as bactérias foram encontradas isoladas; o segundo conjunto, representado a verde, é constituído por 8 imagens em que as bactérias presentes se encontravam organizadas em pares; o terceiro subconjunto, composto pelas restantes 32 imagens, inclui agrupamentos maiores e mais complexos destes microrganismos.

**Tabela 4.1:** Parâmetros de validação do algoritmo de segmentação proposto calculados para 50 imagens.

	$S_A$	$S_M$	$FN$	$FP$	$TP$	$P$ (%)	$S$ (%)	$F-m$ (%)
	1087	1176	115	26	1061	97,608	90,221	93,769
	1249	1428	179	0	1249	100,000	87,465	93,313
	2832	3050	299	81	2751	97,140	90,197	93,540
	3340	3988	750	202	3138	93,952	80,710	86,829
	1375	1276	31	70	1305	94,909	97,680	96,274
	1229	1573	354	1	1228	99,919	77,623	87,371
	2386	2625	260	21	2365	99,120	90,095	94,392
	1958	2394	436	0	1958	100,000	81,788	89,982
	3418	3768	464	114	3304	96,665	87,686	91,957
	2631	2869	263	25	2606	99,050	90,833	94,594
	1136	1159	53	30	1106	97,359	95,427	96,383
	2771	3050	380	101	2670	96,355	87,541	91,737
	1056	1216	161	1	1055	99,905	86,760	92,870
	1282	1418	179	43	1239	96,646	87,377	91,778
	1020	1101	82	1	1019	99,902	92,552	96,087
	1586	1607	108	87	1499	94,515	93,279	93,893
	1489	1610	136	15	1474	98,993	91,553	95,127
	3108	3584	487	11	3097	99,646	86,412	92,558
	1735	2018	285	2	1733	99,885	85,877	92,353
	2449	2798	358	9	2440	99,633	87,205	93,006
	1259	1352	128	35	1224	97,220	90,533	93,757
	1750	1814	126	62	1688	96,457	93,054	94,725
	1581	1424	31	188	1393	88,109	97,823	92,712
	2118	2426	333	25	2093	98,820	86,274	92,121
	2159	2761	645	43	2116	98,008	76,639	86,016
	2256	2731	495	20	2236	99,113	81,875	89,673
	1457	1352	30	135	1322	90,734	97,781	94,126
	1671	1737	206	140	1531	91,622	88,140	89,847
	2147	2426	338	59	2088	97,252	86,068	91,319
	1799	2164	367	2	1797	99,889	83,041	90,689
	2062	2320	293	35	2027	98,303	87,371	92,515
	4188	4380	508	316	3872	92,455	88,402	90,383
	2110	2197	181	94	2016	95,545	91,761	93,615
	2308	2470	261	99	2209	95,711	89,433	92,465
	2212	2031	26	207	2005	90,642	98,720	94,509
	3180	3216	243	207	2973	93,491	92,444	92,964
	2582	2617	273	238	2344	90,782	89,568	90,171
	2311	2808	554	57	2254	97,534	80,271	88,064
	3765	4327	585	23	3742	99,389	86,480	92,486
	2639	3204	636	71	2568	97,310	80,150	87,900
	4416	5289	938	65	4351	98,528	82,265	89,665
	4663	5277	725	111	4552	97,620	86,261	91,590
	2367	2808	484	43	2324	98,183	82,764	89,816
	2997	3266	355	86	2911	97,130	89,130	92,959
	2650	3091	508	67	2583	97,472	83,565	89,984
	5521	6572	1121	70	5451	98,732	82,943	90,151
	4785	5277	682	190	4595	96,029	87,076	91,334
	3695	4327	659	27	3668	99,269	84,770	91,449
	3992	4656	725	61	3931	98,472	84,429	90,911
	7938	9053	1384	269	7669	96,611	84,712	90,271



Com base nos valores de precisão (%), sensibilidade (%) e *F-measure* (%) adquiridos para cada uma das imagens consideradas, calculou-se a média deste conjunto de parâmetros para cada um dos três grupos discriminados na **Tabela 4.1** e ainda para o conjunto global (isto é, considerando as 50 imagens). Os valores obtidos através desta etapa encontram-se discriminados na **Tabela 4.2**, abaixo destacada.

**Tabela 4.2:** Resumo dos valores dos parâmetros de validação do algoritmo de segmentação proposto obtidos para cada grupo de imagens avaliado.

NÚMERO DE IMAGENS	PRECISÃO (%)	SENSIBILIDADE (%)	<i>F-MEASURE</i> (%)
10	97,836	87,430	92,202
8	97,158	91,264	94,059
32	96,533	87,068	91,397
50	<b>96,887</b>	<b>87,797</b>	<b>91,920</b>

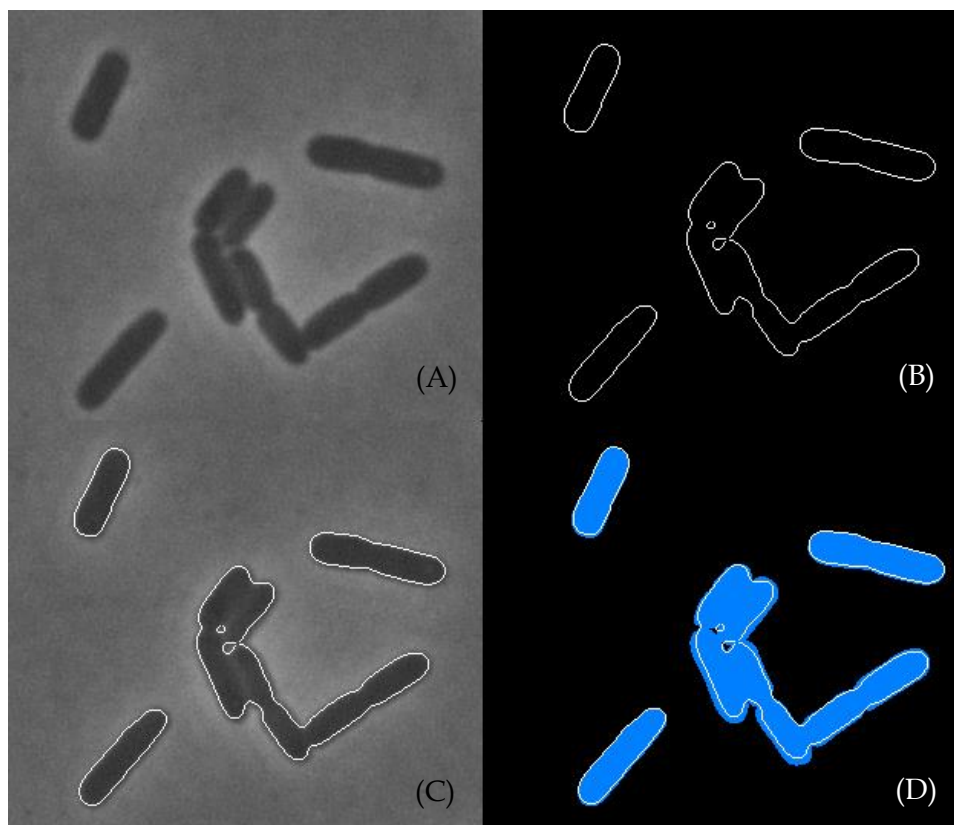
Do mesmo modo que se gerou a **Tabela 4.2** para validar o algoritmo desenvolvido no âmbito desta tese, gerou-se de forma paralela e simultânea uma segunda tabela que permitiu comparar os resultados obtidos, exatamente para o mesmo conjunto de exemplos considerado, para um método de segmentação automático desenvolvido pelo LBD. A média da precisão obtida através deste último método foi de 90,796%, a da sensibilidade 94,540% e a do parâmetro *F-measure* 92,096%. Estes valores correspondem a uma diferença na precisão das classificações igual de 6,091%, na sensibilidade igual a 6,743% e no *F-measure* igual a 0,176%.

Apesar das diferenças nos valores da sensibilidade e *F-measure* não serem muito alarmantes, procurou-se compreender a razão destas discrepâncias e das diferenças evidentes entre a classificação do algoritmo proposto comparativamente à gerada pela segmentação de referência (ou seja, a segmentação manual).

Ao avaliar-se os resultados obtidos ao longo das diversas etapas do método de segmentação proposto verificou-se que, para inúmeras bactérias, havia uma discordância significativa (relativamente à segmentação manual) no que diz respeito à morfologia assumida pelos dos contornos dos microrganismos. Praticamente em todas as imagens, verificou-se que os contornos extraídos através do método de segmentação manual (e consequentemente, também os contornos associados ao algoritmo automático) eram bastante mais largos do que os obtidos através do presente algoritmo.

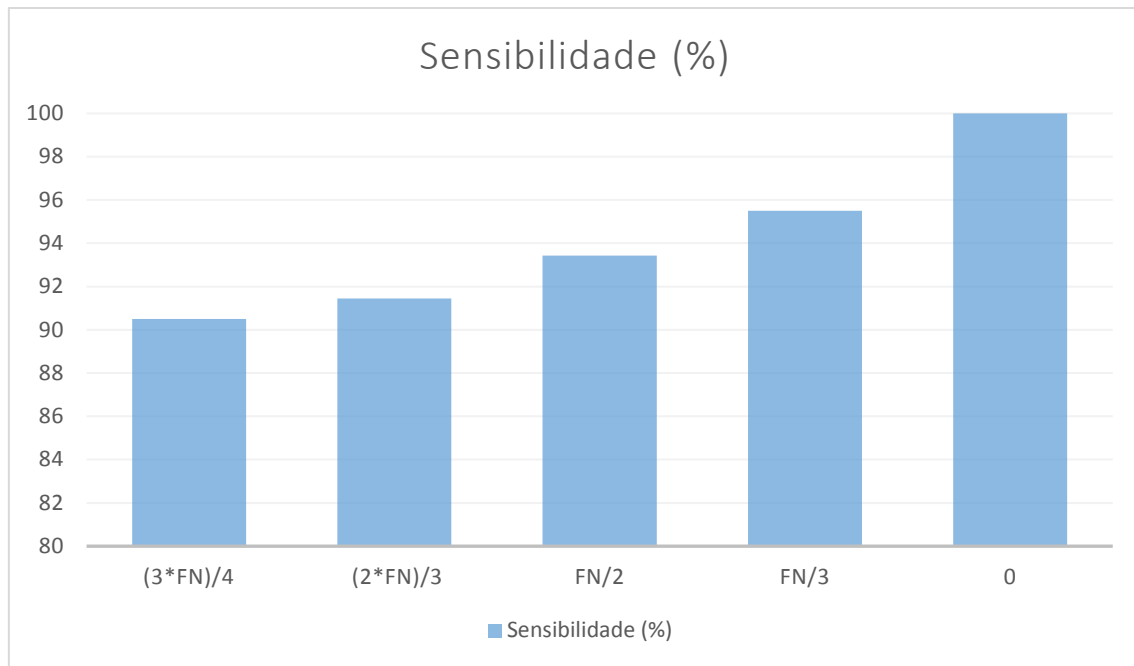
Na da **Figura 4.2**, posteriormente apresentada, é possível observar-se essa diferença. Na parte superior da figura encontra-se (à esquerda) a imagem original e (à direita) os contornos

obtidos através do novo método de segmentação proposto. Na parte inferior da figura, para comparação, estes contornos foram sobrepostos à imagem original (à esquerda) e à imagem segmentada manualmente (à direita).

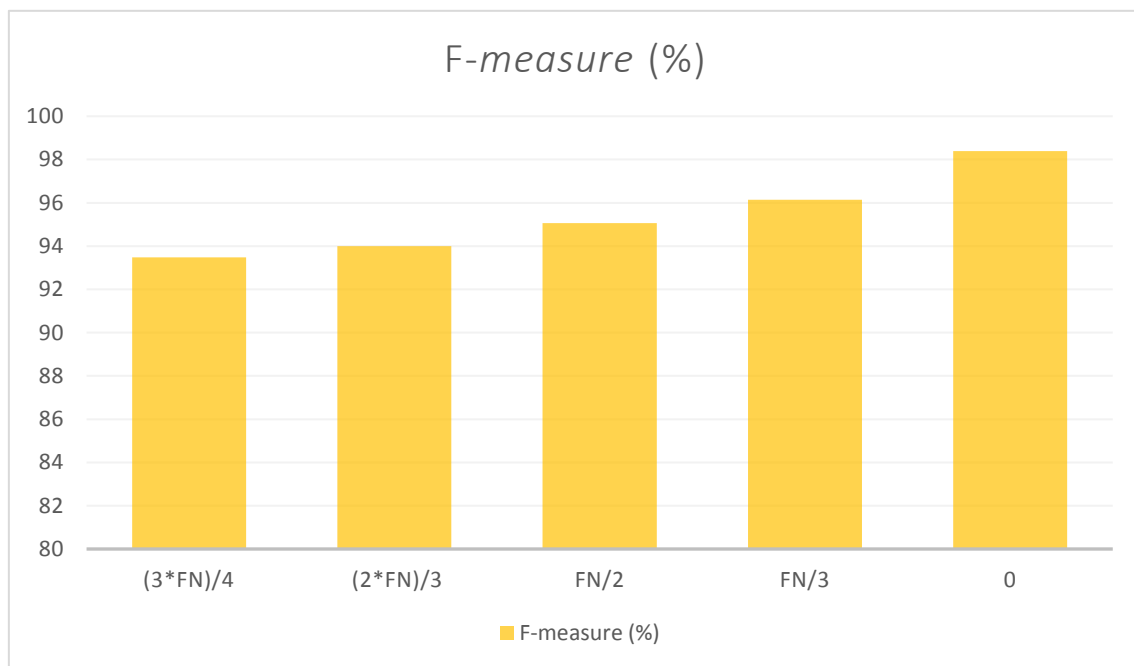


**Figura 4.2:** (A) Imagem original; (B) Contornos das bactérias presentes na imagem original extraídos a partir do método apresentado na secção 3.3.3; (C) Sobreposição da imagem original com os contornos extraídos; (D) Sobreposição da segmentação manual com os contornos extraídos.

No que diz respeito aos parâmetros discriminados na **Tabela 4.1**, a discrepância nas morfologias dos contornos influencia fortemente o valor dos falsos negativos. De modo a simular crescentes níveis de aproximação entre a morfologia obtida manualmente e a morfologia obtida através do algoritmo proposto, consideraram-se quatro tipos de correção em que FN seria reduzido e passaria a assumir um valor equivalente a:  $\frac{3*FN}{4}$ ,  $\frac{2*FN}{3}$ ,  $\frac{FN}{2}$ ,  $\frac{FN}{3}$  e 0. Ao aplicar-se cada um destes ajustes obteve-se, respetivamente, um valor médio de sensibilidade igual a 90.506%, 91.454%, 93.423%, 95.497% e 100% e um valor médio do parâmetro *F-measure* igual a 93.485%, 94.002%, 95.055%, 96.138% e 98.397% - valores bastante mais satisfatórios do que os obtidos anteriormente. A variação do valor médio da sensibilidade e da *F-measure*, em função do tipo de ajuste aplicado, encontram-se respetivamente representados nos gráficos da **Figura 4.3** e **Figura 4.4**.



**Figura 4.3:** Comparação da sensibilidade obtida para o algoritmo proposto considerando diferentes tipos de aproximações entre a morfologia dos contornos obtidos pelo algoritmo desenvolvido e a morfologia dos contornos definida pelos resultados da segmentação manual.



**Figura 4.4:** Comparação da sensibilidade obtida para o algoritmo proposto considerando diferentes tipos de aproximações entre a morfologia dos contornos obtidos pelo algoritmo desenvolvido e a morfologia dos contornos definida pelos resultados da segmentação manual.

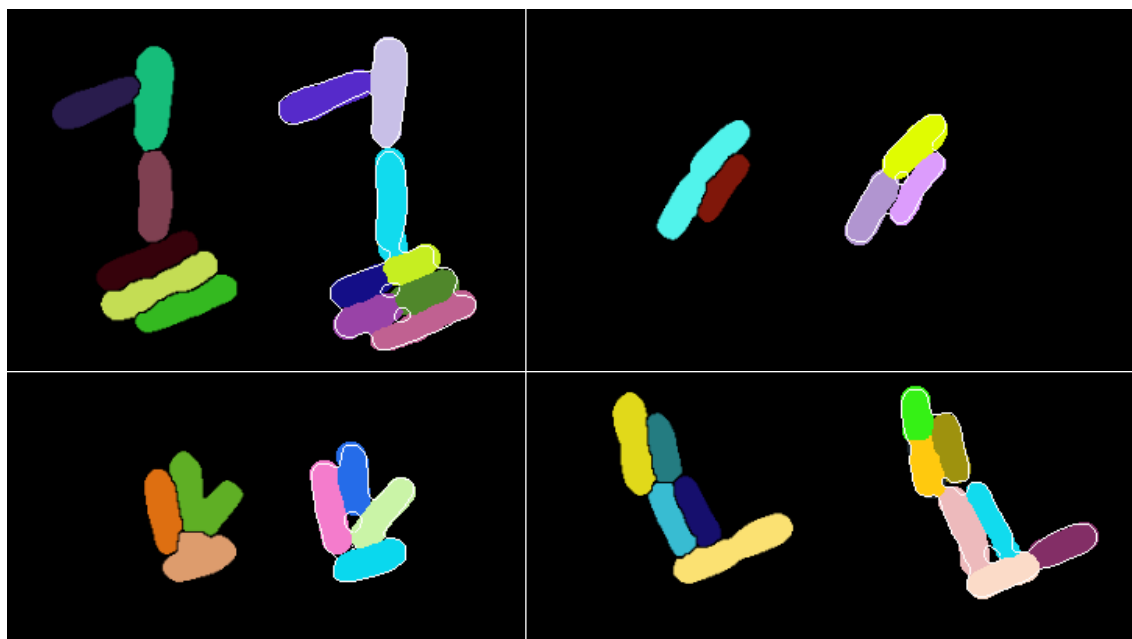
## 4.3 Análise de resultados

O algoritmo de segmentação desenvolvido, tendo em consideração o conjunto de imagens testado, apresenta uma precisão média de classificação igual a 96,887%, uma sensibilidade média igual a 87,797% e um valor de *F-measure* médio igual a 91,920%. Estes valores, obtidos para cada uma das imagens analisadas, encontram-se discriminados na **Tabela 4.1**. Entre os diferentes tipos de agrupamento de bactérias considerados, verificou-se que a eficiência da classificação obtida (combinando os três parâmetros de validação analisados) foi consideravelmente maior para os casos em que as bactérias se encontravam agrupadas em pares.

Ao comparar-se a eficiência do algoritmo implementado com a de um segundo algoritmo (automático) desenvolvido pelo LBD para segmentar o mesmo tipo de imagens, verificou-se que a precisão deste último era (em 6,091%) inferior à do algoritmo desenvolvido, a sensibilidade (em 6,743%) superior e o valor de *F-measure* praticamente o mesmo (sendo apenas superior em 0,176%). No entanto, apesar do algoritmo automático possuir um elevado número de píxeis coincidentes com o resultado da segmentação manual, inspecionando visualmente os resultados obtidos, verificou-se que numa grande parte das bactérias não se encontravam corretamente segmentadas.

Na **Figura 4.5**, são apresentados quatro conjuntos de bactérias segmentados. Para cada exemplo considerado, à esquerda é possível visualizar-se o resultado da segmentação automática e à direita o resultado obtido através da aplicação do método de segmentação desenvolvido. Observando cada caso discriminado na figura é possível concluir que a aplicação do algoritmo automático conduz, em muitos dos casos, à subsegmentação das imagens. Na prática, isto significa que, através deste método, são detetadas menos bactérias do que aquelas que realmente existem. Através do método de segmentação desenvolvido, por ser necessário indicar o centro de cada bactéria presente na imagem, este problema é fortemente atenuado.

Ao detetar-se uma discrepância entre a morfologia dos contornos das bactérias obtidos através do método de segmentação proposto e da segmentação de referência, avaliou-se o efeito que uma série de ajustes exerceria na eficiência do algoritmo. O objetivo destas correções seria aproximar cada vez mais as morfologias dos contornos detetados através destes dois métodos. Verificou-se que, consoante o tipo de ajuste aplicado, o valor médio do parâmetro *F-measure* (o qual mede a qualidade do algoritmo) poderia aumentar até 98.397%. Esta correlação pode ser analisada através no gráfico da **Figura 4.4**.



**Figura 4.5:** Comparação entre alguns dos resultados de segmentação obtidos pelo algoritmo automático desenvolvido pelo LBD (para cada exemplo, à esquerda) e os resultados obtidos através da aplicação do algoritmo de segmentação proposto no âmbito desta tese (para cada exemplo, à direita).

Apesar dos resultados obtidos serem bastante satisfatórios existem três fatores principais que podem afetar consideravelmente a eficiência do algoritmo proposto:

1. A elevada diversidade de tamanhos e morfologias das bactérias apresentadas na imagem;
2. A iluminação insuficiente e um elevado nível de desfocagem das imagens;
3. O local onde é efetuado o clique central por parte do utilizador;

Dos três parâmetros mencionados anteriormente, aquele mais afeta os resultados obtidos através do novo método de segmentação proposto é o terceiro. Quando as bactérias se encontram isoladas ou até mesmo aglomeradas em pares, pequenas variações no local onde o clique central é executado não alteram significativamente os resultados obtidos. No entanto, para maiores agrupamentos do que estes, um clique mal executado poderá afetar não só a segmentação da bactéria indicada mas também a segmentação do conjunto de bactérias adjacentes a esta. Como tal, é fulcral o utilizador estar consciente desta restrição e procurar selecionar as bactérias clicando tão mais próximo do centro de cada microrganismo quanto possível. Apesar deste inconveniente, o método proposto demonstrou, através de estatísticas e da inspeção visual, numa grande parte dos

casos, ser mais eficiente do que o algoritmo automático paralelamente analisado. Comparativamente ao manual, a eficiência do algoritmo desenvolvido, atualmente, é consideravelmente menor. No entanto, o número de interações necessárias passou para metade e, não exigindo outro tipo de ajustes por parte do utilizador, o tempo necessário para a segmentação de cada imagem conseguiu ser reduzido de horas ou (em alguns casos) inclusivamente dias para apenas alguns minutos. Através da aplicação de alguns ajustes futuros na morfologia dos contornos detetados pelo presente algoritmo seria possível aumentar significativamente o nível de compatibilidade da classificação destas duas técnicas.



## Conclusão

### 5.1 Considerações finais

O principal objetivo associado à elaboração desta tese consistiu na criação de um novo método de segmentação a ser aplicado a um conjunto alargado de imagens microscópicas da *E. coli*. Estas imagens foram obtidas por microscopia de contraste de fase e facultadas pelo LBD (*Laboratory of Biosystem Dynamics*) da *Tampere University of Technology*.

O procedimento de segmentação desenvolvido baseou-se num conjunto de informações, conhecidas *a priori*, que permitiram descrever geometricamente a morfologia destas bactérias. Com base nestes dados, desenvolveu-se um algoritmo semi-automático que, para cada microrganismo presente na imagem, procurou preencher os seus contornos através de uma sequência finita de círculos. O primeiro círculo gerado para cada *E. coli* é inicializado, através de um clique, num dado ponto da imagem indicado pelo utilizador como sendo o centro da bactéria. Posteriormente, as dimensões desta figura geométrica são ajustadas bem como o posicionamento do seu centro. Depois desta etapa estar concluída, inicia-se um processo cíclico através do qual novos círculos (de igual dimensão) são sucessivamente gerados e ajustados até que se preencha por completo o interior de todas as bactérias identificadas. De modo a distinguir cada microrganismo presente na imagem e posteriormente avaliar a qualidade da segmentação, atribuiu-se uma cor distinta a cada um deles.

Visando a atenuar o ruído que se sobrepunha inicialmente às imagens de estudo, aplicou-se um filtro de média como método integrante da etapa de pré-processamento das imagens. De seguida, de modo a extrair-se os contornos das bactérias presentes nas imagens, efetuou-se uma

binarização da mesmas (baseada no método de Otsu) seguida de uma convolução com um filtro Laplaciano.

De modo a facilitar a aplicação do algoritmo proposto, projetou-se e desenvolveu-se uma interface gráfica do utilizador (GUI). Para além da hipótese de se proceder à segmentação das bactérias, neste ambiente foram incluídas diversas ferramentas de processamento e análise de imagens. O conjunto de elementos incluídos nesta interface encontra-se discriminado no capítulo 3.1.

Ao aplicar-se o algoritmo desenvolvido a um conjunto composto por 50 imagens distintas, obteve-se uma percentagem de êxito (medida pelo parâmetro *F-measure*) igual a 91,920%. Após ter-se avaliando visualmente os resultados de segmentação obtidos, este valor revelou-se bastante aquém das expectativas. Depois de se investigar a causa desta discrepância descobriu-se que o problema consistia em diferentes interpretações daqueles que tinham sido considerados os contornos das bactérias. Ao simular-se uma aproximação entre estes limites, previu-se que a eficiência do algoritmo pudesse aumentar até 98.397%.

Comparativamente à segmentação manual, esta técnica releva-se bastante menos demorada e intensiva do ponto de vista da interação requerida por parte do utilizador. Comparativamente a outros métodos automáticos desenvolvidos, como por exemplo o método considerado na análise de resultados, esta técnica veio solucionar, na maioria dos casos, o problema frequente da sub-segmentação das imagens.

Um dos principais fatores que condicionam a qualidade da segmentação obtida através do algoritmo desenvolvido relaciona-se com a posição do clique inicial efetuado pelo utilizador. Caso se selecione, através desse evento, um ponto da imagem que significativamente distante do centro da bactéria (tanto longitudinalmente como transversalmente) o resultado da segmentação da mesma bem como o do conjunto em que esta se insere, pode ficar seriamente comprometido. Para além deste fator, caso a direção definida inicialmente como a de propagação dos círculos na bactéria não seja a correta e os ajustes não forem suficientemente abrangentes para corrigir este erro, o resultado da segmentação pode ser semelhantemente afetado.

Ao longo deste projeto foram identificados alguns pontos que poderiam ser melhorados por intermédio de alguns desenvolvimentos futuros seguidamente apresentados.



## 5.2 Desenvolvimentos futuros

Um dos pontos fulcrais a ser trabalhado no algoritmo proposto consiste em aproximar o processo da detecção dos contornos das bactérias desenvolvido à metodologia atualmente adotada pelo LBD. Através deste ajuste, deixaria de existir uma discrepância tão grande entre os contornos obtidos através dos dois métodos e por isso, consequentemente, o número de píxeis coincidentes iria aumentar. Esta alteração, tal como mencionado anteriormente, traduzir-se-ia, por si só, num aumento considerável na eficiência da segmentação.

O processo associado à propagação dos círculos na bactéria é um procedimento que depende fortemente da ordem através da qual cada bactéria do conjunto selecionado vai sendo preenchida e da direção de propagação inicialmente definida para cada microrganismo. Este processo, embora funcione bastante bem para conjuntos mais reduzidos de bactérias, quando aplicado a conjuntos significativamente maiores e mais complexos, por vezes falha. Assim sendo, seria necessário futuramente otimizar-se a ordem pela qual as bactérias vão sendo preenchidas bem como o processo através do qual se define a direção de propagação inicial da bactéria. À partida, através deste melhoramento, seria possível eliminar ou, pelo menos, reduzir bastante a quantidade de ajustes necessários, o que se traduziria numa diminuição do tempo de processamento associado a este método de segmentação.

Por último, na interface, deveriam ser implementadas e introduzidas duas novas funcionalidades: uma delas, que possibilitasse (de forma cíclica) selecionar um dado conjunto de bactérias e visualizar o resultado da segmentação desse agrupamento até que o utilizador desse por terminada a tarefa; a segunda, que permitisse anular o último resultado de segmentação obtido. Estes dois métodos, aplicados alternadamente, permitiriam corrigir eventuais erros de segmentação gerados para um determinado grupo de bactérias evitando que essa segmentação errónea pontual afete o resultado da segmentação global da imagem.



# Bibliografia

- [1] Reshes, G., Vanounou, S., & Feingold, M. (2008). Cell Shape Dynamics in *Escherichia coli*. *Biophysical Journal*, 251-264.
- [2] *ThermoFisher Scientific*. (2017, Julho 16). Retrieved from <https://www.thermofisher.com/pt/en/home/life-science/cell-analysis/single-cell-analysis.html>
- [3] Ribeiro, A. S. (2017, Junho 21). *Laboratory of Biosystem Dynamics (LBD)*. Retrieved from <http://www.cs.tut.fi/~sanchesr/>
- [4] Wang, D., & Bodovitz, S. (2010). Single cell analysis: the new frontier in 'omics'. *Trends in Biotechnology*, 28, 281-290.
- [5] Essential Knowledge Briefing. (2015). *Live-Cell Imaging*. USA: John Wiley & Sons, Inc. Obtido de Live-Cell Imaging.
- [6] *Nikon Instruments - Microscopes and Imaging Systems*. (2017, Julho 20). Retrieved from Techniques: Time-Lapse: [https://www.nikoninstruments.com/en\\_EU/Learn-Explore/Techniques/Time-Lapse](https://www.nikoninstruments.com/en_EU/Learn-Explore/Techniques/Time-Lapse)
- [7] FAO Animal Production and Health Division. (2011). A review of *Escherichia coli* as an emerging food-borne pathogen. *EMPRES Transboundary Animal Diseases Bulletin*, pp. 20-27.
- [8] Welch, R. A. (2006). The Genus *Escherichia*. *Prokaryotes*, 60-71.
- [9] *Atlas da Saúde*. (2017, Junho 10). Retrieved from Infecções por bacilos: <http://www.atlasdasaude.pt/publico/content/infecoes-por-bacilos>
- [10] Bentley, R., & Meganathan, R. (2001). Biosynthesis of menaquinone (menaquinose) vitamin K2 and ubiquinone (coenzyme Q): a perspective on enzymatic mechanisms. *Vitam. Horm*, 61, 173-218.
- [11] Bush, L. M., & Perez, M. T. (2017, Junho 13). *Manuais MDS*. Retrieved from Infecções por *Escherichia coli*: <http://www.msmanuals.com/pt/profissional/doen%C3%A7as-infeciosas/bacilos-gram-negativos/infec%C3%A7%C3%B5es-por-escherichia-coli>
- [12] Bossolan, N. R. (2002). *Introdução à Microbiologia*. Instituto de Física de São Carlos: IFSC/LCE.
- [13] Reshes, G., Vanounou, S., & Feingold, M. (2008). Cell Shape Dynamics in *Escherichia coli*. *Biophysical Journal*, 251-264.
- [14] Geoffrey, R. E. (2007). *A Célula: Uma Abordagem Molecular*. Boston University: Artmed.
- [15] Mannheimer, W. A. (2002). *Microscopia dos Materiais: Uma Introdução*. Rio de Janeiro: E-papers Serviços Editoriais.
- [16] Stephens, D. J., & Allan, V. J. (2003). Light microscopy techniques for live cell imaging. *Science (New York, N.Y.)*, 300 (5616), 82–86.
- [17] KASVI. (2017, Fevereiro 23). Retrieved from Microscópio: Conheça as diversas técnicas de microscopia: <http://www.kasvi.com.br/microscopio-tecnicas-microscopia/>
- [18] Murphy, D. B. (2001). *Fundamentals of Light Microscopy and Electronic Imaging*. Canada: Wiley-Liss, Inc.

- [19] Murphy, D., Oldfield, R., Schwartz, S., & Davidson, M. (2017, Julho 3). *MicroscopyU*. Retrieved from Introduction to Phase Contrast Microscopy: <https://www.microscopyu.com/techniques/phase-contrast/introduction-to-phase-contrast-microscopy>
- [20] Salvetti, A. R. (2008). *A História da Luz*. São Paulo: Livraria da Física.
- [21] Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., & Walter, P. (2007). *Molecular Biology of the Cell*. New York: Garland Science.
- [22] Smochina, C. (2011). Image Processing Techniques and Segmentation Evaluation.
- [23] Meijering, E. (2012). Cell Segmentation: 50 Years Down the Road. *IEEE Signal Processing Magazine*, 140-145.
- [24] Tolles, W. E. (1955). The Cytoanalyzer - An example of physics in medical research. *Transactions of the New York Academy of Sciences*, 3, 250-156.
- [25] Prewitt, J. M., & Mendelsohn, M. L. (1966). The analysis of cell images. *Annals of the New York Academy of Sciences*, 1035-1053.
- [26] Meijering, E. (2010). Neuron tracing in perspective. *Cytometry Part A*, 77, 693-704.
- [27] Webb, R. H. (1996). Confocal optical microscopy. *Rep. Prog. Phys.*, 59, 427-471.
- [28] Bengtsson, E., Wahlby, C., & Lindblad, J. (2004). Robust cell image segmentation methods. *Pattern Recognition and Image Analysis*, 14, 157-167.
- [29] Teeffelen, S., Shaevitz, J. W., & Gitai, Z. (2012). Image analysis in fluorescence microscopy: Bacterial dynamics as a case study. *Bioessays*, 34(5): 427-436.
- [30] MathWorks. (2017, Junho 24). Retrieved from <https://www.mathworks.com/discovery/image-segmentation.html>.
- [31] Efford, N. (2000). Digital Image Processing: A Practical Introduction Using 21Java. *Pearson Education*.
- [32] Wang, D., & Bodovitz, S. (2010). Single cell analysis: the new frontier in 'omics'. *Trends in Biotechnology*, 28, 281-290.
- [33] Wu, Q., Merchant, F. A., & Castleman, K. R. (2008). *Microscope Image Processing*. Burlington: MA.
- [34] Filho, O. M., & Neto, H. V. (1999). *Processamento Digital de Imagens*. Rio de Janeiro: Brasport.
- [35] DPI - Divisão de Processamento de Imagens. (2017, Julho 17). Retrieved from Teoria: Processamento de Imagens: <http://www.dpi.inpe.br/spring/teoria/filtrage/filtragem.htm>
- [36] GTA/UFRJ Grupo de Teleinformática e Automação. (2017, Julho 6). Retrieved from Crescimento de Regiões por Agregação de Píxeis: [https://www.gta.ufrj.br/grad/07\\_2/eliseu/CrescimentodeRegiesporAgregaodePíxeis.htm](https://www.gta.ufrj.br/grad/07_2/eliseu/CrescimentodeRegiesporAgregaodePíxeis.htm)  
1
- [37] Wu, Q., Merchant, F. A., & Castleman, K. R. (2008). *Microscope Image Processing*. Burlington: MA.
- [38] Araujo, A. F. (2010). *Novas Metodologias para Análise de órgãos e músculos*. Porto: Universidade do Porto - Faculdade de Engenharia.
- [39] Gonzalez, R. C., & Woods, R. E. (1993). *Digital Image Processing*. Reading, MA: Addison-Wesley.
- [40] CCWJ. (2017, Julho 6). Retrieved from Image Analysis: Morphological Operations: <https://sites.ualberta.ca/~ccwj/teaching/image/morph/>

- [41] Moreira, N. (2017, Junho 13). *Faculdade de Ciências da Universidade do Porto*. Retrieved from <https://www.dcc.fc.up.pt/~nam/aulas/0001/pi/trabalho2/trab2/>
- [42] *Cockahoop*. (2017, Agosto 3). Retrieved from Pantone vs CMYK vs RGB: <http://cockahoop-digital.co.uk/pantone-vs-cmyk-vs-rgb/>
- [43] Saleh, B. E., & Teich, M. C. (1991). *Fundamentals of Photonics*. New York: John Wiley & Sons.
- [44] Pássaro, N. M., & Faria, N. M. (2006). *Filtragem de Imagens Fixas*. FEUP.
- [45] *Matlabtricks*. (2017, Agosto 27). Retrieved from A tutorial on intensity transforms of images: <http://matlabtricks.com/post-45/a-tutorial-on-intensity-transforms-of-images>
- [46] *Natural Resources Canada*. (2017, Junho 18). Retrieved from Image Enhancement: <http://www.nrcan.gc.ca/earth-sciences/geomatics/satellite-imagery-air-photos/satellite-imagery-products/educational-resources/9389>
- [47] *ACM SIGGRAPH*. (2017, Junho 26). Retrieved from CGSD - Gamma Correction Explained: [https://www.siggraph.org/education/materials/HyperGraph/color/gamma\\_correction/gamma\\_intro.html](https://www.siggraph.org/education/materials/HyperGraph/color/gamma_correction/gamma_intro.html)
- [48] Lyshevski, S. E. (2003). *Engineering and Scientific Computations Using MATLAB*. New Jersey: John Wiley & Sons, Inc.
- [49] Junior, D. P., Nakamiti, G. S., Engelbrecht, A. M., & Bianchi, F. (2011). *Algoritmos e Programação de Computadores*. Rio de Janeiro: Elsevier.
- [50] *FM2S*. (2017, Junho 29). Retrieved from O que é um fluxograma?: <http://www.fm2s.com.br/tudo-sobre-fluxograma/>
- [51] *Citisystems*. (2017, Outubro 7). Retrieved from Fluxograma de Processo - O que é, como elaborar e benéficos: <https://www.citisystems.com.br/fluxograma/>
- [52] *MathWorks*. (2017, Junho 28). Retrieved from Create Apps with Graphical User Interfaces in MATLAB: <https://www.mathworks.com/discovery/matlab-gui.html>
- [53] Hunt, B. R., Lipsman, R. L., & Rosenberg, J. M. (2014). *A Guide to MATLAB for Beginners and Experienced Users*. United Kingdom: Cambridge University Press.
- [54] *SDSU Library & Information Access*. (2017, Junho 15). Retrieved from Image Processing Toolbox - Intensity Adjustment: <https://edoras.sdsu.edu/doc/matlab/toolbox/images/enhanc12.html>
- [55] Gonzalez, R. C., Woods, R. E., & Eddins, S. L. (2009). *Digital Image Processing using MATLAB*. Gatesmark Publishing.
- [56] *Stackoverflow*. (2017, Junho 1). Retrieved from Increase image contrast using look up table in MATLAB: <https://stackoverflow.com/questions/25191218/increase-image-contrast-using-look-up-table-in-matlab>
- [57] Olaya, V. (2017, Agosto 21). *Sistemas de Información Geográfica*. Retrieved from Procesado de imágenes: <http://volaya.github.io/libro-sig/chapters/Imagenes.html#Filtros de suavizado>
- [58] *Electronics and Computer Science*. (2017, Julho 3). Retrieved from Median Filtering: [http://users.ecs.soton.ac.uk/msn/book/new\\_demo/median/](http://users.ecs.soton.ac.uk/msn/book/new_demo/median/)
- [59] Rebaza, J. (2017, Julho 16). *Universidad Nacional de Trujillo*. Retrieved from Detección de bordes mediante el algoritmo de Canny: [file:///C:/Users/slukasvo/Downloads/Deteccion\\_de\\_bordes\\_mediante\\_el\\_algoritmo\\_de\\_Canny.pdf](file:///C:/Users/slukasvo/Downloads/Deteccion_de_bordes_mediante_el_algoritmo_de_Canny.pdf)

- [60] *Monfort University*. (2017, Agosto 7). Retrieved from Laplacian/Laplacian of Gaussian: <http://www.cse.dmu.ac.uk/~sexton/WWWPages/HIPR/html/log.html>
- [61] *What-When-How*. (2017, Julho 30). Retrieved from Edge Detection (Image Processing) : <http://what-when-how.com/embedded-image-processing-on-the-tms320c6000-dsp/edge-detection-image-processing-part-1/>
- [62] Sanches, C. H., Fontoura, P. J., Vieira, P. F., & Batista, M. A. (2015). Técnicas de Suavização de Imagens e Eliminação de Ruídos. *Anais do EATI - Encontro Anual de Tecnologia da Informação*, 21-30.
- [63] Carneiro, T. (2017, Agosto 12). *Instituto de Computação - UFF*. Retrieved from Segmentação: <http://www2.ic.uff.br/~aconci/curso/binari~3.htm>
- [64] Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9, 62-66.
- [65] *Wikimedia Commons*. (2017, Junho 15). Retrieved from Area moment of inertia of a circular sector: [https://commons.wikimedia.org/wiki/File:Area\\_moment\\_of\\_inertia\\_of\\_a\\_circular\\_sector.svg](https://commons.wikimedia.org/wiki/File:Area_moment_of_inertia_of_a_circular_sector.svg)
- [66] *Florida Center for Instructional Technology*. (2017, Junho 5). Retrieved from Unit Circle Marked At Special Angles: [http://etc.usf.edu/clipart/43200/43206/unit-circle14\\_43206.htm](http://etc.usf.edu/clipart/43200/43206/unit-circle14_43206.htm)